

A Thesis Submitted for the Degree of PhD at the University of Warwick

Permanent WRAP URL:

<http://wrap.warwick.ac.uk/133710>

Copyright and reuse:

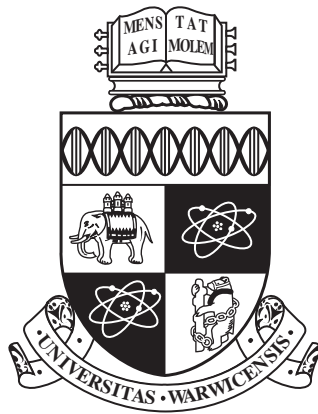
This thesis is made available online and is protected by original copyright.

Please scroll down to view the document itself.

Please refer to the repository record for this item for information to help you to cite it.

Our policy information is available from the repository home page.

For more information, please contact the WRAP Team at: wrap@warwick.ac.uk



**Nowcasting User Behaviour with Social Media
and Smart Devices on a Longitudinal Basis:
From Macro- to Micro-level Modelling**

by

Adam Tsakalidis

A thesis submitted to The University of Warwick

in partial fulfilment of the requirements

for admission to the degree of

Doctor of Philosophy

Supervised by: Prof. Alexandra I. Cristea and Dr. Maria Liakata

Department of Computer Science

The University of Warwick

September 2018

Abstract

The adoption of social media and smart devices by millions of users worldwide over the last decade has resulted in an unprecedented opportunity for NLP and social sciences. Users publish their thoughts and opinions on everyday issues through social media platforms, while they record their digital traces through their smart devices. Mining these rich resources offers new opportunities in sensing real-world events and indices (e.g., political preference, mental health indices) in a longitudinal fashion, either at the macro (population)-, or at the micro(user)-level.

The current project aims at developing approaches to “nowcast” (predict the current state of) such indices at both levels of granularity. First, we build natural language resources for the static tasks of sentiment analysis, emotion disclosure and sarcasm detection over user-generated content. These are important for opinion monitoring on a large scale. Second, we propose a general approach that leverages textual data derived from generic social media streams to nowcast political indices at the macro-level. Third, we leverage temporally sensitive and asynchronous information to nowcast the political stance of social media users, at the micro-level using multiple kernel learning. We then focus further on the micro-level modelling, to account for heterogeneous data sources, such as information derived from users’ smart phones, SMS and social media messages, to nowcast time-varying mental health indices of a small cohort of users on a longitudinal basis. Finally, we present the challenges faced when applying such micro-level approaches in a real-world setting and propose directions for future research.

Dedicated to my family.

Acknowledgements

First of all, I would like to thank my supervisors, Prof. Alexandra Ioana Cristea and Dr. Maria Liakata. Your guidance and feedback on my research over these years have been rather inspiring, teaching me the most important lesson of my Ph.D. – learning how to learn. I have been extremely fortunate to have you as my supervisors and I will always feel grateful for your continuous support. Thank you for everything you have done for me.

I would also like to thank all of my co-authors and all those many people with whom we have exchanged ideas over our research during these four years. I feel especially grateful to Dr. Theo Damoulas, whose feedback on many aspects of my work has been rather motivational for me to advance my research further and open my eyes to new concepts and ideas. Also, I would like to thank Prof. Stephen Jarvis, Prof. Rob Procter and Dr. Jane Sinclair for the great feedback they provided me with on my annual review sessions, helping me re-think and re-shape my research based on their constructive comments.

Thankfully, I have also been rather fortunate to have many good friends outside my work, who have all contributed to the current Thesis, even without knowing what “NLP” stands for. The time we spend together is a vital element of my life and my work. Thank you all for being around.

Last but not least, I would like to thank my parents, Iraklis and Zoe, and my sister, Anatoli, for their love and the great support they have provided me with during all these many years. I will always owe the world to you.

Finally, special thanks to Maria for supporting and tolerating me, even at these very last and stressful moments of my studies. I will always be grateful for that, among many others.

Declarations

This thesis is submitted to the University of Warwick in support of my application for the degree of Doctor of Philosophy. It has been composed by myself and has not been submitted in any previous application for any degree. The work presented (including data generated and data analysis) was carried out primarily by the author. Parts of this thesis have been published as full papers by the author:

- Tsakalidis, A., Aletras, N., Cristea, A.I. and Liakata, M., 2018, October. Nowcasting the Stance of Social Media Users in a Sudden Vote: The Case of the Greek Referendum. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (pp. 367-376). ACM. [240]
- Tsakalidis, A., Liakata, M., Damoulas, T. and Cristea, A.I., 2018, September. Can we assess mental health through social media and smart devices? Addressing bias in methodology and evaluation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 407-423). Springer, Cham. [241]
- Tsakalidis, A., Papadopoulos, S., Voskaki, R., Ioannidou, K., Boididou, C., Cristea, A.I., Liakata, M. and Kompatsiaris, Y., 2018. Building and evaluating resources for sentiment analysis in the Greek language. *Language Resources and Evaluation*, 52(4), pp.1021-1044. [245]
- Tsakalidis, A., Liakata, M., Damoulas, T., Jellinek, B., Guo, W. and Cristea, A., 2016. Combining Heterogeneous User Generated Data to Sense Well-being. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers* (pp. 3007-3018). [242]

-
- Tsakalidis, A., Papadopoulos, S., Cristea, A.I. and Kompatsiaris, Y., 2015. Predicting Elections for Multiple Countries using Twitter and Polls. *IEEE Intelligent Systems*, 30(2), pp.10-17. [243]

The author has also contributed to the following works, albeit not directly linked to this thesis:

- Zubiaga, A., Voss, A., Procter, R., Liakata, M., Wang, B. and Tsakalidis, A., 2017. Towards Real-time, Country-level Location Classification of World-wide Tweets. *IEEE Transactions on Knowledge Data Engineering*, 29(9), pp.2053-2066. [262]
- Wang, B., Liakata, M., Tsakalidis, A., Kolaitis, S.G., Papadopoulos, S., Apostolidis, L., Zubiaga, A., Procter, R. and Kompatsiaris, Y., 2017. TOTEMSS: Topic-based, Temporal Sentiment Summarisation for Twitter. *Proceedings of the IJCNLP 2017, System Demonstrations*, pp.21-24. [253]
- Townsend, R., Tsakalidis, A., Zhou, Y., Wang, B., Liakata, M., Zubiaga, A., Cristea, A. and Procter, R., 2015. Warwickdcs: From Phrase-based to Target-specific Sentiment Recognition. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)* (pp. 657-663). [238]

Sponsorship and Grants

The research presented in this thesis was made possible by the support of EPSRC through the University of Warwick's Centre for Doctoral Training in Urban Science and Progress (grant EP/L016400/1) and through The Alan Turing Institute (grant EP/N510129/1).

Abbreviations

ANew Affective Norms for English Words

API Application Programming Interface

AVG A model that always predicts the user's average mood score

CB1 Count-Based approach by Tumasjan et al. [247] to predict election results

CB2 Count-Based approach, partly used by Sang and Bos [216] to predict election results

CBOW Continuous Bag Of Words

cos Cosine Similarity

CPB Counting-Poll-Based approach to predict election results

DA DeviceAnalyzer

DATE A model that is trained on the timestamp of the mood score to predict a user's mental health

EBL Emoticon-Based Lexicon

EU European Union

FEAT A model trained on smart phone-derived and social media features to predict a user's mental health score, based on the setting in [36]

FF Feed Forward

fn False Negative

fp False Positive

GP Gaussian Process

GrAFS Greek Affect and Sentiment lexicon

GRGE Greek General Elections (dataset)

idf Inverse Document Frequency

JSON JavaScript Object Notation

KBL Keyword-Based Lexicon

LASSO Least Absolute Shrinkage and Selection Operator

LAST A model that always predicts the user's last entered mood score

LIWC Linguistic Inquiry and Word Count

LOIOCV Leave-one-instance-out cross-validations

LOUOCV Leave-one-user-out cross-validations

LR Linear/Logistic Regression

MA Moving Average

MAE Mean Absolute Error

MC Majority Class

MCKL Multiple Convolution Kernel Learning

MKL Multiple Kernel Learning

MP Election forecasts by MetaPolls

MSE Mean Squared Error

MSOL Macquarie Semantic Orientation Lexicon

NCE Noise Contrastive Estimation

NHS National Health Service

NLP Natural Language Processing

NN Neaural Network

NRC National Research Council

OSM Online Social Media

PANAS Positive and Negative Affect Scale

PB Poll-Based approach to predict election results

PCA Principal Component Analysis

PERS Personalised setup

PHQ Patient Health Questionnaire

PMI Pointwise Mutual Information

POS Part-Of-Speech

PW Election forecasts by PollWatch

RAND A model that is trained on random features to predict a user's mental health

RBF Radial Basis Function

ReLU Rectified Linear Unit

RF Random Forest

RMSE Root Mean Squared Error

RNN Recursive Neural Network

SA Sentiment Analysis

SB Approach by Sang and Bos [216] to predict election results

SG Skip-Gram

SMO Sequential Minimal Optimisation

SMS Short Message Service

SVM Support Vector Machine

SVR Support Vector Regression

t-SNE t-Distributed Stochastic Neighbor Embedding

TDF Thessaloniki Documentary Festival (dataset)

tf Term Frequency

TIFF Thessaloniki International Film Festival (dataset)

tn True Negative

tp True Positive

TPB Twitter-Poll-Based approach to predict election results

UNIQ Unique setup

URL Uniform Resource Locator

WEMWBS Warwick-Edinburgh Mental Well-Being Scale

Contents

Abstract	ii
Dedication	iii
Acknowledgements	iv
Declarations	v
Sponsorship and Grants	vii
Abbreviations	viii
List of Figures	xx
1 Introduction	1
1.1 Overview	1
1.2 Research Questions and Objectives	3
1.3 Challenges	8
1.4 Thesis Outline	11
I Background	13
2 Technical and Theoretical Background	14
2.1 Overview	15
2.2 Data Aggregation from Social Media	15
2.2.1 Twitter	15
2.2.2 Twitter Streaming API	16
2.3 Text Representation	17
2.3.1 Pre-processing	17
2.3.2 Feature Extraction	18

2.4	Algorithms	23
2.4.1	Classification Algorithms	23
2.4.2	Regression Algorithms	31
2.5	Evaluation	34
2.5.1	Validation Approaches	34
2.5.2	Evaluation Metrics	35
2.6	Summary and Conclusion	37
3	Related Work	38
3.1	Monitoring User Behaviour through Social Media and Smart De- vices: Overview	39
3.2	Sentiment Analysis	39
3.2.1	Generating Sentiment Resources (preliminary analysis) . .	42
3.3	Social Media and Elections (RQ1, RQ2)	43
3.3.1	Social Media and Elections at the Macro-level (RQ1) . . .	44
3.3.2	Social Media and Elections at the Micro-level (RQ2) . . .	47
3.4	Mental Health and Digital Media (RQ3)	50
3.4.1	Correlation Tasks	50
3.4.2	Stand Alone User or Text Classification	51
3.4.3	Longitudinal Models for Assessing Mental Health	53
3.5	Ethical Considerations	56
II	Macro-level Modelling Using Social Media	59
4	Building and Evaluating Sentiment Analysis Resources	60
4.1	Introduction	61
4.2	Generating the Resources	62
4.2.1	GrAFS Lexicon Creation	62
4.2.2	Twitter-Specific Sentiment Lexicons	65
4.2.3	Twitter-Specific Word Embeddings	69
4.3	Experimental Setup	70

4.3.1	Datasets	70
4.3.2	Feature Extraction	72
4.3.3	Classification and Regression Algorithms	74
4.4	Results	75
4.4.1	Task 1: Sentiment Analysis	75
4.4.2	Task 2: Emotion Intensity Analysis	77
4.4.3	Task 3: Sarcasm Detection	78
4.4.4	Key Findings	81
4.5	Summary and Conclusion	83
5	Macro-level Modelling Using Social Media	84
5.1	Introduction	85
5.2	Background: The EU Elections	86
5.3	Methodology	86
5.3.1	Data Aggregation	87
5.3.2	Modelling	87
5.3.3	Sentiment Analysis	89
5.3.4	Algorithms	91
5.4	Data	91
5.4.1	Twitter	91
5.4.2	Opinion Polls	92
5.5	Results	93
5.6	Post-hoc Analysis and Discussion	97
5.7	Summary and Conclusion	99
III	Micro-level Modelling Using Social Media and Smart Devices	101
6	Micro-level Modelling Using Social Media	102
6.1	Introduction	103
6.2	The Greek Bailout Referendum	104

6.3	Task Description	105
6.4	Data	106
6.4.1	Training Set	107
6.4.2	Test Set	109
6.5	Models	111
6.5.1	Convolution Kernels	111
6.5.2	Text Kernels	112
6.5.3	Network Kernels	112
6.5.4	Kernel Summation	113
6.5.5	SVMs with Convolution Kernels	113
6.5.6	Multiple Convolution Kernel Learning (MCKL)	113
6.6	Experimental Setup	114
6.6.1	Features	114
6.6.2	Models	116
6.6.3	Evaluation	117
6.7	Results	118
6.7.1	Nowcasting Voting Intention	118
6.7.2	Robustness Analysis	120
6.8	Qualitative Analysis	121
6.8.1	Language	121
6.8.2	Network	123
6.9	Summary and Conclusion	126
7	Micro-level Modelling Using Heterogeneous Data Sources	128
7.1	Introduction	129
7.2	A Dataset of Heterogeneous Textual and Mobile phone data . . .	131
7.3	Methodology	134
7.3.1	Data matrix creation and Features	134
7.3.2	Baseline definition	136
7.3.3	Experiments and Models	139

7.4	Evaluation and Results	141
7.5	Summary and Conclusion	144
8	Challenges in Micro-level Modelling Using Heterogeneous Data Sources	146
8.1	Introduction	147
8.2	Problem Statement	149
8.2.1	Training on past values of the target (LOIOCV, LOUOCV)	152
8.2.2	Inferring Test Labels (LOIOCV)	153
8.2.3	Predicting Users (LOUOCV)	155
8.3	Experiments	156
8.3.1	Datasets	156
8.3.2	Task Description	158
8.3.3	Features	159
8.4	Results	161
8.4.1	P1: Using Past Labels	161
8.4.2	P2: Inferring Test Labels	162
8.4.3	P3: Predicting Users	164
8.5	Proposal for Future Directions	167
8.6	Summary and Conclusion	170
9	Proposal for Future Directions in Micro-level Modelling Using Heterogeneous Data Sources	171
9.1	Introduction	172
9.2	Methods	172
9.2.1	Behavioural Embeddings: <code>mobile2vec</code>	173
9.3	Training <code>mobile2vec</code>	175
9.4	Empirical Validation	177
9.4.1	Number of epochs	177
9.4.2	Dimensionality of Embeddings	179
9.5	Discussion	181

9.6	Summary and Conclusion	182
IV	Conclusion	184
10	Conclusions	185
10.1	Main Findings	186
10.1.1	Preliminary (document-level) Analysis	186
10.1.2	RQ1: Macro-level Monitoring Using Social Media	187
10.1.3	RQ2: Micro-level Monitoring Using Social Media	187
10.1.4	RQ3: Micro-level Monitoring Using Heterogeneous Sources	188
10.2	Directions for Future Research	190
10.2.1	Document-level Analysis	191
10.2.2	Macro-level Monitoring Using Social Media	191
10.2.3	Micro-level Monitoring Using Social Media	192
10.2.4	Micro-level Monitoring Using Heterogeneous Sources . . .	193
	Appendices	219
A	Guidelines for the Manually Annotated Lexicon	220
A.1	Aim	220
A.2	Mining Triantafyllidis Lexicon	220
A.3	Annotation Guidelines: Subjectivity and Sentiment	220
A.4	Annotation Guidelines: Emotions	221
A.5	Lexicon Use	221
B	List of Keywords	222

List of Figures

1.1	Annual online connectivity statistics in the UK, as recorded by the Office for the National Statistics (https://www.ons.gov.uk).	3
2.1	High-level overview of the macro- or micro-level monitoring process using social media and smart devices.	15
4.1	Distributions (in log scale) of word scores before (blue) and after (green) the morphological expansion (from top-left to bottom-right: subjective, positive, negative, angry, disgust, fear, happy, sad, surprise).	66
5.1	Number of political tweets aggregated per day in the three electoral races, after a 7-day MA filter.	92
5.2	MAE per training window size for different algorithms and countries.	98
6.1	Number of tweets in Greek per hour. The period highlighted in red indicates the nine evaluation time points, starting before the announcement of the referendum and ending in the day before the date of the referendum.	107
6.2	Number of nodes N (users), re-tweet edges E and average degree ($ E / N $) of the re-tweeting network over time, in a cumulative fashion (blue) and in a sliding window approach (red) of network construction (i.e., based on the re-tweeting activity of the past seven days).	116
6.3	Macro-average F-score across all evaluation days using TEXT , NETWORK and BOTH user representations.	118

6.4	Change in performance (mean/standard deviation) compared to the results in Figure 6.3, after 100 experiments with added noisy features.	120
6.5	Scores of n-grams related to the political parties/leaders, pre (18/06-26/06) and post (27/06-05/07) the referendum announcement. Scores<0 (>0) indicate that n-grams appear mostly in tweets of NO (YES) voters.	122
6.6	Difference in cosine similarity ($\cos_{post}(w_{no/yes}, w) - \cos_{pre}(w_{no/yes}, w)$) between the <i>no/yes</i> (red/blue) word vectors $w_{no/yes}$ and each of their most similar words in the two periods.	124
6.7	Network representations of YES/NO (blue/red) users, before (left) and after (right) the referendum announcement.	125
6.8	Normalised difference of similarity of YES/NO (blue/red) users in our modelling (left) and in a sliding window approach (right). . .	126
7.1	Average and standard deviations of the mood form scores obtained by the 19 subjects.	135
7.2	Geo-visual projection of the subjects' visited locations. Each colour indicates a unique user and the size of their spot indicates the number of unique GPS samples at that location.	137
7.3	Actual VS Predicted charts for the best performing algorithm (RF) on the three targets.	143
7.4	Feature set weights in RF (red) and MKL (blue) for the positive and the well-being targets, training on all features in the per-user normalisation approach.	144
8.1	The three types of evaluations found in literature. Train instances (or users, in <i>LOUOCV</i>) are coloured blue; test instances (users) are coloured red.	148

8.2	Moving averages filter (right) applied to the raw “positive” scores (left) of a randomly selected user. The smoothing effect helps in making the long-term mood trend clearer, in exchange for missing the instant mood score.	154
8.3	Average and standard deviation mood scores (y-axis) on a per-subject basis (x-axis) for the three targets (positive, negative, wellbeing) in our dataset.	156
8.4	P2: Sensitivity/specificity (blue/red) scores over the {positive, negative, wellbeing, stress} targets by training on different time windows on the <i>LOIOCV</i> (top) and <i>LOUOCV</i> (bottom) setups, similar to [36].	163
8.5	P3: Actual vs predicted charts for the “positive” and “wellbeing” targets in <i>LOIOCV</i> . The across-subjects R^2 is negative.	165
9.1	Loss per 500 iterations, using different number of negative samples (lines) and latent space dimensionality (above: [10, ..., 50]; below: [60, ..., 100]).	177
9.2	Visualisation of the (20-dim) embeddings, using different number of epochs in our training (5, 20, 50, 100).	178
9.3	Average euclidean distance between semantically related feature categories, per epoch. Different charts correspond to different latent feature dimensionality (5, 10, 30, 40, 50, 60, 70, 90, 100).	179
9.4	Visualisation of the embeddings, using different dimension on the resulting matrices in our training (5, 20, 50, 100) and training with one negative sample over 20 epochs.	180
9.5	Average euclidean distance between semantically related feature categories, per latent feature dimensionality. Different charts correspond to different epochs used during training (1, 5, 15, 25, 50, 100).	181

List of Tables

1.1	Number of monthly active users (based on https://www.statista.com) of some of the most highly (Alexa) ranked online social networks.	3
1.2	Summary of data collected, annotated and published during this Ph.D. (per research question).	10
2.1	Summary of the evaluation metrics used in this Thesis.	37
3.1	Works on classifying social media posts/users with respect to mental health conditions.	52
3.2	Works on predicting mental health in a longitudinal manner. . .	54
4.1	Annotators' agreement for subjectivity (Pearson Correlation), positive and negative (Cohen's Kappa), respectively.	64
4.2	Annotators' agreement (Pearson Correlation) for the six emotions.	64
4.3	Number of tweets per-class in the sentiment analysis task.	71
4.4	F-measure based on 10-fold cross-validation for Task 1.	76
4.5	F-measure based on cross-domain experiments for Task 1. The first column indicates the test dataset, after training the models on the rest.	77
4.6	MSE for the Emotion Prediction task (Task 2), using 5-fold cross validation.	79
4.7	Pearson correlation for the Emotion Prediction task (Task 2), using 5-fold cross validation.	80
4.8	Cross-emotion results for Task 2.	80
4.9	F-score on the Sarcasm Detection Task.	81
5.1	Variance of reported shares in the processed polls.	93

5.2	Comparison between the performance of our approach against various baselines (Germany).	95
5.3	Comparison between the performance of our approach against various baselines (The Netherlands).	95
5.4	Comparison between the performance of our approach against various baselines (Greece).	95
5.5	Comparison between the performance of our approach against various baselines, macro-averaged across the three electoral races.	96
6.1	Political position, austerity, referendum stance and national election result (January 2015) of the political parties that are used as seeds in our modelling.	108
6.2	Number of users (u) and tweets (t) used in our experiments per evaluation day.	109
6.3	Average F-score and standard deviation across all evaluation days using TEXT , NETWORK and BOTH user representations. SVM _s and SVM _{st} denote the SVM with convolution kernels (SVM _w , SVM _n) and (SVM _{wt} , SVM _{nt}), respectively.	119
6.4	Most similar words to YES and NO (translated to English), when training word2vec on different time periods.	122
6.5	Examples of highly re-tweeted tweets after the announcement of the referendum.	123

7.1	R^2 root mean squared error (ϵ) of the different models based on the three feature sets (DA, TEXT, ALL) and with respect to the three different ground truth scores (positive, negative, well-being). Values for both setups with respect to the user normalisation (with and without) are presented. The index used in the R^2 column indicates the (i) final and (ii) per-user normalisation of the best-performing setup (n for normalisation, s for standardisation, $-$ for none). Only the final normalisation method (i) is indicated in experiments performed without per-user normalisation.	142
8.1	Summary of the three evaluation frameworks.	148
8.2	Presentation of <i>Dataset 1</i> in numbers by source (rows) and type of item (columns). Note that all messages are private.	157
8.3	Summary of experiments. The highlighted settings indicate the settings used in the original papers; “Period” indicates the period before each mood form completion during which the features were extracted.	159
8.4	P1: Results following the approach in [134]. The AVG baseline outperforms the LR model in <i>LOUOCV</i> , consistently. If the features derived from previously self-reported target scores are not used (-mood), the performance drops even more.	162
8.5	P2: Performance (sensitivity/specificity) of the SVM classifier trained over 14 days of smartphone/social media features (FEAT) compared against 3 naïve baselines.	164
8.6	P3: Results following the evaluation setup in [242] (<i>MIXED</i>), along with the results obtained in the <i>LOIOCV</i> and <i>LOUOCV</i> settings with (+) and without (-) per-user input normalisation.	166
8.7	P3: Accuracy by following the evaluation setup in [103] (<i>MIXED</i>), along with the results obtained in <i>LOIOCV</i> & <i>LOUOCV</i>	166
9.1	Parameters used in <i>mobile2vec</i> training process.	176

CHAPTER 1

Introduction

1.1 Overview

The wide adoption of World Wide Web and its services over the past decade has provided online users with the ability to generate and share content in unprecedented volumes through their connected devices (see Figure 1.1). Online social media – defined by [115] as “a group of Internet-based applications that build on the ideological and technological foundations of Web 2.0, and that allow the creation and exchange of user-generated content” – form a vital element of communication among billions of online users (see Table 1.1), who are using them to generate and share information on everyday matters, potentially acting in this sense as “online social sensors” of real-world events and opinions towards them. Establishing effective approaches on mining knowledge out of these social media data streams can enable us to predict (or “*nowcast*”¹) the current state of a population or an environment at different levels of granularity [12, 124, 214, 26, 126]. This is nowadays becoming more feasible, since different sources of asynchronous and heterogeneous information, such as data derived from users’ smart phones and sensors, start becoming available. Such information can accompany the users’ social media data to provide a more complete overview of the behaviour of the user and his/her preferences.

The current Ph.D. Thesis explores different approaches on *using longitudinal data derived from social media and smart devices for the task of nowcasting real-world indices*, such as political- or mental health-related indices. The key

¹The terms “predict” and “nowcast” are often used interchangeably in this Thesis. “Predict” is intended to refer to the task of applying a pre-trained machine learning model that “predicts” a real-world index; if the model is applied based on the current state of some temporally varying input, then it essentially “nowcasts” that real-world index.

components of the modelling process lie in the areas of *natural language processing* (NLP) and *machine learning*, with an explicit focus on longitudinal modelling, aiming to create methods that are applicable to real-world problems and can generalise across different cases. Depending on the granularity of the analysis, such tasks are divided into two broad categories:

- (a) **macro-level** approaches aim at leveraging large and user-agnostic streams of data to nowcast the current state of a real-world index at the population-level; examples in this category include the tasks of predicting final election results or building country-level mental health indices based on streams of user-generated content;
- (b) **micro-level** approaches on the other hand aim at leveraging user-specific data to nowcast the current state of this single user; examples include the tasks of nowcasting the voting intention of a social media user using his/her social media posts or nowcasting the current mental health state of a specific subject using data derived from his/her social media account and his/her smart phone.

This distinction is important, primarily owed to the different nature of the performed task: in *macro-level* modelling, we aim at nowcasting a general index covering a whole population. Such approaches benefit from the availability of large-scaled resources (aggregated data streams of social media posts); however, these resources are not representative of the whole population [156] and thus they often lack the ability to generalise over different cases, such as a different population [78, 150] or points in time [208]. Furthermore, such approaches are often vulnerable to bias due to the presence of online bots [93]. Finally, modelling user-generated content in a fine-grained macro-level spatial resolution (e.g., ward- or city-level) demands aggregating large-scale user-generated data from thousands of online users, which in turn can be challenging, owed to the lack of availability of reliable and representative geo-tagged data streams in such resolution [183]. On the contrary, in *micro-level* tasks we aim at now-

casting a single user’s index. Hence, by nature, such approaches fail to provide an aggregate prediction of a social or urban index. However, the output of such models is of crucial importance in several cases. For example, the task of nowcasting mental health indices using digital media trails of some users is more crucial in the micro-level than in the macro-level, due to its ability to provide insights on needed interventions tailored to the needs of specific users. Similarly, in the political domain, a campaign strategist might want to adjust his/her political campaign towards specific users, which is impossible to achieve at the macro-level.

The effective modelling at both levels of granularity provides the opportunity to study the human behaviour at the large scale and mine knowledge indicative of the current state of an individual or a population. In this sense, a key challenge presented in this Thesis is to build models that can generalise across different cases and be applicable to real-world problems. This is of huge importance, in order to provide to stakeholders with the ability to incorporate such methods in their workflow.

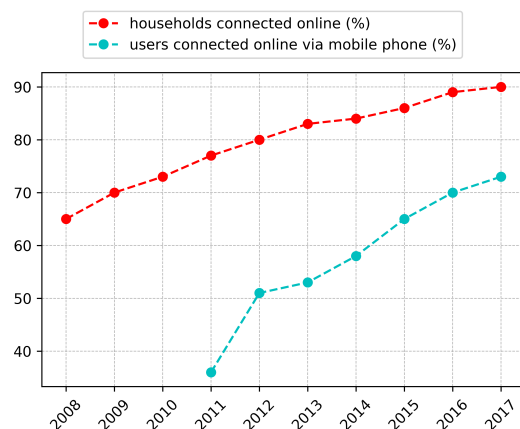


Figure 1.1: Annual online connectivity statistics in the UK, as recorded by the Office for the National Statistics (<https://www.ons.gov.uk>).

	Active	Alexa
Name	Users	Rank
Facebook	2.20B	3
Twitter	0.34B	10
Instagram	1.00B	13
Sina Weibo	0.41B	20
LinkedIn	0.29B	28

Table 1.1: Number of monthly active users (based on <https://www.statista.com>) of some of the most highly (Alexa) ranked online social networks.

1.2 Research Questions and Objectives

The main objective of the research presented in this Thesis is to exploit data derived from social media and smart phones – either of a whole population or of a single user – in a longitudinal fashion, aiming to nowcast the preferences or the current state of the involved entities – either of the whole population (*macro-level*) or of the single user (*micro-level*). To achieve this demands a breakdown of the objectives into several individual research questions.

Given that one of the main types of communication in social networks is the written text, establishing appropriate methods for extracting information out of the textual user-generated content is highly important for any further large-scale analysis at the macro-level. A key task that often needs to be resolved in this context is the task of *sentiment analysis* [181] (classifying a piece of text as being “positive”, “negative” or “neutral”); other related tasks are the *emotion (affect) analysis* [161] and *sarcasm detection* [88]. A major problem of traditional approaches in sentiment analysis is their lack of the ability to generalise over different domains [244]. For example, a model that has been trained on social media posts discussing about politics cannot be effectively applied in posts discussing about sports without adaptation (a process known as “transfer learning” or “domain adaptation” [22]). To this end, generating NLP resources for constructing effective text representations that can be successfully integrated in machine learning algorithms across different domains is of high importance and a first step towards monitoring opinion in the large scale.

Such *preliminary analysis on the document-level* of social media posts can be helpful on monitoring real-world indices, when large streams of such documents are aggregated and analysed in a temporal fashion. Thus, we define our first research question as follows:

RQ1. *Can we use data streams from social media in order to nowcast real-world indices on the macro-level?*

To address RQ1, we focus on the task of predicting electoral outcomes using data aggregated from a social network platform (i.e., Twitter). The effective modelling of this specific task can provide to stakeholders (e.g., politicians, campaign strategists) with the ability to analyse and predict the current political state of a population on the large scale. Ideally, to demonstrate the effectiveness of such a method, we need to provide a final prediction for an electoral race *before* the announcement of the actual results and test its effectiveness in other electoral cases. We set our objectives related to RQ1 as follows:

- O1a** extract features from social media posts on a temporal basis that can be indicative of a political party’s popularity;
- O1b** use them to train time-series models, using opinion polls as the macro-level ground truth index;
- O1c** demonstrate the ability of the proposed method to be employed in the real-world by (a) providing an unbiased prediction before the announcement of the results, (b) working in multiple electoral cases and (c) comparing its accuracy against state-of-the-art approaches currently used by political scientists.

By nature, such macro-level modelling tasks help stakeholders gain an overall overview of the population preferences or state, but fail to provide insights on how to tailor their actions towards specific users. However, as discussed previously, micro-level modelling of a certain group of social media users is often more beneficial to stakeholders in the real world. To tackle this issue in a longitudinal manner, we pose the following question:

RQ2. *Can we use data streams from a specific group of social media users in order to nowcast their real-world indices at the micro-level?*

To answer this, we again focus on the political domain, aiming to nowcast the voting intention of a group of (manually annotated) social media users. We focus on a particularly interesting case of a sudden electoral case, in which the macro-level, time-series models that were previously discussed cannot be applied due to the short time period of the electoral race. Building accurate and robust models for predicting the voting intention of social media users on a temporal basis is of particular interest to campaign strategists, who cannot rely on macro-level models under such a time-constrained setting. To this end, we set the following objectives for our first micro-level modelling task:

- O2a** build accurate and robust models leveraging temporally sensitive information about social media users for the task of nowcasting their intention under a time-constrained setting;
- O2b** provide quantitative and qualitative insights on the appropriateness of the proposed approach.

Besides social media, the last decade has seen a continuous increase in the adoption of smart devices connected online, such as smart phones (see Figure 1.1), by the public. Such carry-on devices record various aspects of a user's behaviour, such as the location that he/she is at, the number of calls that he/she receives and their duration, etc., in a longitudinal fashion. This heterogeneous information can accompany information derived from social media for various micro-level tasks. Furthermore, in micro-level tasks we are often interested in monitoring a non-static index of a user, such as his/her physical or mental health, which changes on a daily basis. Here we pose the following question:

RQ3. *Can we use asynchronous and heterogeneous data streams from a specific group of users in order to nowcast their temporally sensitive real-world indices at the micro-level?*

We study RQ3 under the task of nowcasting the mental health of a group of subjects. In contrast to the case of political stance where the target is relatively stable, here we are interested in the challenge of studying a micro-level task under a much more dynamically time-varying target, which we aim at predicting. Furthermore, the information that is available through smart devices is intuitively better suited to the health rather than the political domain. Our objectives related to RQ3 are the following:

- O3a** build accurate models, leveraging heterogeneous information derived from the smart phones and social media accounts of a group of subjects, to nowcast their time-varying mental health state in a longitudinal fashion;
- O3b** provide explanatory model predictions, demonstrating the information that is mostly predictive of the users’ mental health state.
- O3c** apply these models *under a real-world setting* to test their ability to be employed as real-world applications.

Our O3b objective is highly important if we aim at building models that can be employed as applications in a real-world setting. For the same reason, it is crucial to study RQ3 under such a real-world setting, as stated by O3c. A common challenge of micro-level tasks that leverage sensitive data (e.g., data derived from the subjects’ smart phones) is that currently they often employ a small number of subjects. This creates several issues with respect to the ability of such models (e.g., micro-level mental health predictors) to generalise and be applicable under a pragmatic setting. However, this ability to generalise is crucially important *(a)* for ensuring, even empirically, real-world deployment and *(b)* for providing insights on the types of user behaviour that affect the real-world index under study. For this reason, we examine closely our O3c objective, by replicating state-of-the-art approaches in the domain of nowcasting mental health with smart devices and social media, demonstrating their weaknesses and proposing directions for future work.

By addressing the research questions presented in this section, we aim at mining and studying user behaviour at different levels of granularity, with a primary focus on *longitudinal* modelling and the applicability of the proposed methods under a *real-world* setting.

1.3 Challenges

We list some of the major challenges posed throughout the tasks tackled in this Thesis, as follows:

C1 Generating resources appropriate to the problem under study is the first necessary step towards addressing each of the research questions set in this Thesis. Albeit not a key methodological part of the presented research, aggregating large-scale content and generating high-quality (and, often, manually annotated) datasets appropriate for the task in hand are two crucial steps with respect to model evaluation at the latter stage.

C2 Temporal modelling of heterogeneous and asynchronous noisy information is particularly challenging from a methodological perspective. Dealing with user-generated textual content is especially challenging, due to its informal and noisy nature. Extracting meaningful representations for the task at hand out of such noisy text is a fundamental task that needs to be tackled effectively. Furthermore, integrating heterogeneous and asynchronous information (e.g. smart phone data, network information) with purely text-based approaches in a temporal fashion poses further challenges, especially for addressing **RQ2–RQ3**. Most of the traditional approaches in machine learning build the training examples (aka “instances”) in a static and feature aggregate fashion, thus losing the temporal information and the separability of different information sources, respectively. Current state-of-the-art approaches in various temporal/sequential modelling tasks employ deep learning approaches, such as Recursive Neural Networks (RNNs) and their extensions [96]. However, such ap-

proaches demand many thousands of labeled instances to train on, which may not be readily available at the user level in a real-world setting. To this end, **training models on not-so-big data**, while incorporating different information sources in a temporal fashion, is a problem that needs to be effectively tackled if we aim at providing solutions to a real-world macro- or micro-level online monitoring task.

C3 Working under a real-world setting for any macro- or micro-level task is of crucial importance, since we aim at developing approaches that can be employed under such a setting. As we will show later on this Thesis, this seemingly obvious challenge is often not addressed appropriately in past work [134, 24, 36, 104, 103]. Model building under a real-world setting is highly important not only for providing real-world solutions, but also for assessing the model’s performance in an unbiased setting.

We effectively tackle each of these challenges in different tasks throughout this Thesis. To tackle **C1**, we collect and annotate the following datasets, enabling us to study the corresponding research questions in a high quality setting (for a summary, refer to Table 1.2): *(a)* we generate datasets consisting of social media posts for the preliminary tasks (“PR” in Table 1.2) of sentiment analysis and sarcasm detection; we develop manually a sentiment and affect lexicon for an under-resource language; we further develop two automatically generated sentiment lexicons and word embedding representations. *(b)* For **RQ1** (macro-level modelling using social media), we aggregate posts from social media related to the political domain in different electoral races; for our ground-truth, we collect several opinion polls from various sources. *(c)* For **RQ2** (micro-level modelling using social media), we collect a large stream of public social media posts and we manually annotate 2.7K social media users with respect to their political stance. *(d)* Finally, for **RQ3** (micro-level modelling using heterogeneous information sources), we aggregate data collected from the smart phones and social media accounts of 30 subjects over several months; to generate

Table 1.2: Summary of data collected, annotated and published during this Ph.D. (per research question).

	Data Collected	Annotated Data	Published Resources
PR	Social media posts (15M)	Lexicon (32K word entries) Sentiment analysis dataset (1,640 posts) Sarcasm detection dataset (2,506 posts)	All "Annotated Data" Word Embeddings (418K word entries) 2 Sentiment Lexicons (191K/32K word entries)
RQ1	Social media posts (1.1M) Opinion Polls (46)	–	Post IDs
RQ2	Social media posts (14.7M)	2.7K social media users (annotated wrt voting intention)	–
RQ3	Social media posts/messages Smart phone and sensor logs Daily responses to psychological tests (30 users)	–	–

our time-varying ground-truth, we collect daily responses of the subjects to two well-established psychological scales.

For **C2**, we propose temporally sensitive models for the macro- and micro-level tasks: (a) for **RQ1**, we employ time-series modelling techniques to model a single information source (i.e., social media posts combined with real-world indices); (b) for **RQ2**, we employ a Multiple Kernel Learning (MKL) approach, by modelling the available time-sensitive and asynchronous information sources derived from social media data via separate convolution kernels; (c) for **RQ3**, we again employ a MKL approach, this time using a different kernel for every available modality derived from a user’s mobile phone and social media activity.

For **C3**, we showcase the ability of our approaches to be employed under a real-world setting or discuss upon their limitations, as follows: (a) for our preliminary (document-level) analysis, we work on the macro- and document-level on multiple real-world datasets in different tasks. (b) For **RQ1**, we build our nowcasting models and showcase their real-world deployment ability by providing their macro-level electoral predictions *before* the announcement of the election results, in order to avoid making potentially biased predictions, as argued in leading past work [150, 78]. (c) For **RQ2**, we work on the micro-level task of nowcasting the users’ voting intention by mimicking a real-world and real-time evaluation setting. (d) Finally, for **RQ3** we follow past evaluation approaches on the micro-level task of assessing mental health indices using heterogeneous data sources. However, by altering our evaluation setup to follow a realistic setting, we observe that the performance of our models drops heavily.

For this reason, in **RQ3/O3c**, we investigate whether such approaches can be employed in a real-world setting; we follow current state-of-the-art approaches and demonstrate their over-optimistic reported performance owed to flaws in the evaluation setup and highlight the crucial importance of setting up a pragmatic evaluation framework for future research.

1.4 Thesis Outline

This Ph.D. Thesis is structured in a macro-to-micro-level modelling manner. Part I provides the necessary background for the comprehension of this Thesis. Then, the analysis of our approaches starts with static (document-level) NLP classification tasks on social media and macro-level modelling of real-world indices using streams of social media data (Part II). Finally, in Part III, we move into the micro-level modelling of users using social media and heterogeneous data, discuss upon their limitations and propose directions for future research. Overall, this Thesis is structured as follows:

- **Part I – Chapter 2** outlines the theoretical background on the topics of natural language processing and machine learning that are necessary for the comprehension of the modelling approaches followed in this Thesis.
- **Part I – Chapter 3** provides a literature review on the macro- and micro-level tasks on the chapters that follow up next.
- **Part II – Chapter 4** focuses on the preliminary document-level modelling of social media streams of data, as a first step towards macro-level modelling. In particular, this chapter presents methods to create NLP resources for the tasks of sentiment analysis, emotion and sarcasm detection over user-generated content of social media. The resources are evaluated in different datasets against standard NLP feature baselines, yielding better performance.
- **Part II – Chapter 5** focuses on **RQ1** and demonstrates how large social

media feeds can be used to model and nowcast real-world indices, with a specific application on the political domain and different electoral cases.

- **Part III – Chapter 6** introduces the micro-level task of nowcasting real-world user-specific indices, using data derived from their social media accounts. As opposed to Chapter 5, the focus here is on the user-level, aiming to nowcast his/her index (which, in this case, is his/her voting preference) on a longitudinal basis, addressing **RQ2**.
- **Part III – Chapter 7** expands the micro-level methodology presented in Chapter 6, to account for *(a)* heterogeneous input data sources (i.e., smart phones, social media) and *(b)* a longitudinal target we aim at predicting (i.e., mental health index), addressing **RQ3**.
- **Part III – Chapter 8** presents the limitations of micro-level modelling when dealing with small-scale datasets, with an application in the task of mental health assessment, and proposes future directions, aiming at the employment of such models in the real-world, as discussed in **RQ3/O3c**.
- **Part III – Chapter 9** presents our steps towards tackling the limitations presented in Chapter 8, with an emphasis on building user representations that map his/her behaviour on a temporally varying latent space, and proposes directions for future research in this field.
- **Part IV – Chapter 10** summarises the key findings and contributions and proposes potential directions for future research.

Part I

Background

CHAPTER 2

Technical and Theoretical Background

The current chapter provides an overview of the research process that has been followed throughout this Thesis. We first present methods on aggregating content from a social media platform (i.e., Twitter). Then, we move into methods for converting a piece of raw text into representations that can be used by a machine learning algorithm. Finally, we briefly present the major machine learning algorithms that have been used in this Thesis and the corresponding evaluation metrics.

2.1 Overview

The typical research process in social or urban monitoring through social media and heterogeneous data streams is demonstrated in Figure 2.1. First, some raw data (in our case, this is primarily textual data) is collected, related to the task we aim at tackling (see Section 2.2). This data is then pre-processed and some $\{x, y\}$ pairs of {features, target} are created (so-called “instances”), based on the pre-processed raw (textual) data that was previously collected and some target y we aim at predicting (e.g., health rates over a population, see Section 2.3). Then, machine learning approaches are incorporated in order to learn a function f based on the $\{x, y\}$ “training” examples that can effectively map some previously unobserved data x^* (“test” example) to their (predicted) target \hat{y} : $\hat{y} = f(x^*)$ (see Section 2.4). Finally, the effectiveness of the model is assessed through various metrics (see Section 2.5).



Figure 2.1: High-level overview of the macro- or micro-level monitoring process using social media and smart devices.

2.2 Data Aggregation from Social Media

2.2.1 Twitter

Twitter¹ is a micro-blogging platform that allows its users to post short messages (a.k.a. “tweets”) of up to 140 characters² to their timelines. During the latest decade, Twitter has a remarkable rise in its numbers, recording 335M monthly active users³ and often serving as the largest source of updates during major

¹<https://twitter.com>

²Since 2017, Twitter allows messages of up to 280 characters for most languages. However, the datasets employed in this Thesis are collected during the period of 2014–2016, thus constrained to the initial 140 characters limit.

³<https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/>

real-world events⁴. The users of this platform are allowed to “follow” other users’ messages, “like” them or “re-tweet” them (that is, share another user’s message, similar to the “share” action of facebook). The short messages mainly contain text, images, URL links, hashtags (i.e., words starting with the # symbol, which, informally, indicates the topic of their message) and user mentions (i.e., another user can be mentioned in a tweet by the use of the @ symbol in the form of “@username”). This short and noisy nature of the messages makes it difficult to extract meaningful knowledge out from them and some pre-processing steps are normally conducted beforehand (see Section 2.3.1).

2.2.2 Twitter Streaming API

Twitter offers a Streaming API⁵, allowing its users to aggregate content shared in real-time within the Twitter platform. Users registered with the Twitter Streaming API are allowed to retrieve tweets in real-time based on three criteria: (a) geo-located tweets within some (max 25) pre-defined locations; (b) tweets coming from (max 5000) pre-specified users; and (c) tweets containing at least one keyword of a pre-specified list of words (max 400 words). If the number of tweets matching a certain query are more than 1% of the overall Twitter volume, then the streaming API will return a random sub-sample of the matched results, of size no more than 1% of its overall volume. This has been proven empirically to be problematic in past work [169] in cases of hitting the 1% limit; however, this is unlikely to be the case in any of the studied cases within this Thesis.

For the purposes of our work, we have used the last option of tweet aggregation (i.e., aggregating tweets based on their content), using various lists of keywords, depending on the performed task. Twitter Streaming API returns a JSON object as a response to our queries, containing various information about the user, his/her profile, the content of the message and other metadata associated with it, such as the location from which the tweet was posted. For the

⁴<https://goo.gl/X3XhWx>

⁵<https://developer.twitter.com/en/docs/tweets/filter-realtime/overview>

most part of this Thesis, we have only used information about the user (i.e., an anonymised user id), the actual content of the tweet and whether it is a tweet that was created by the user posting it or a retweet of another user. Before using the content of the tweet in either a macro- or micro-level task, several pre-processing and feature extraction steps need to be performed, as presented next.

2.3 Text Representation

2.3.1 Pre-processing

Owed to their noisy nature, documents aggregated from online social media posts need a lot of pre-processing before we start extracting features out from them. This pre-processing is essentially formed by some noise reduction steps, aiming to transform a noisy piece of content into a more meaningful document. Here, we list the major pre-processing steps that are commonly used in past work:

- **Lowercasing** aims at convert all documents into their lower case, such that the words “*Lower*” and “*lower*” are mapped to the same concept.
- **Stemming** is a process that converts words such as “*screaming*” and “*screams*” into “*scream*” [197].
- **Stop-word removal** aims at removing some commonly used words (e.g., the words “am”, “is”, etc.), since such words do not carry much of a semantic load.
- **Shortening elongated words** is another step aiming to bring together words such as “*looooool*” and “*loool*”.
- **Replacement of URLs and user mentions** with unique identifiers is commonly used, since these offer little or no information at all for most NLP-related tasks.

- **Replacement of social media-specific symbols**, such as emoticons, with their semantic meaning is a step that aims to map to the same concepts the meanings of “:.)” and “:D” which are used to denote a positive emotion.
- **Non-alphabetic/numeric character removal** aims at removing all symbols that are not alphabetical (or alphanumerical) characters, since those are not useful for most NLP-related tasks.
- **Tokenisation** is a process that splits a sentence into a list of its words; despite seemingly easy, this step is tricky when dealing with user-generated content – Twitter-specific tokenisers have been developed over the last decade to cope with this issue [83].

Part or all of these steps are very commonly used in past work [213, 244, 259]. After this process, each document is represented as a list of words that appear sequentially in it and feature extraction can be performed in a much better structured input.

2.3.2 Feature Extraction

There is a plethora of features that have been derived from text and used in various NLP tasks in past work. Here, we summarize the main types of textual features that will be used in the next chapters. When dealing with heterogeneous data sources (i.e., in our micro-level task using heterogeneous data), the textual features can be accompanied by others which are extracted based on other sources. We will provide the definition of such heterogeneous features in the corresponding chapters (7, 8 and 9).

Ngrams

Given a collection of $|D|$ documents $\{\{w_{00}, \dots, w_{0|D_0|}\}, \dots, \{w_{D0}, \dots, w_{D|D_{D-1}|}\}\}$, the most intuitive representation approach is to create a dictionary, mapping each of the $|W|$ unique sequence of words of length n comprising the documents

in D , into a unique id. Depending on the value of n , we can create unigrams ($n = 1$), bigrams ($n = 2$), trigrams ($n = 3$) etc., or a combination of those. The ids of these “*ngrams*” can then be converted into one-hot-encoded vectors of length $|W|$, resulting into a vector representation for each word, with the value 1 located only at the position of the id of the word (the rest of the entries of this vector have the value 0). Finally, each of the documents $d \in D$ can be represented by a single “*bag-of-words*” vector v_d of length $|W|$, with all of its entries having the value 0, except for the indices that correspond to the ids of the words that comprise d .

This results into creating “*binary*” representations, which fail to provide information about how often a certain ngram appears in a document. For that, we can opt for a weighted scheme to represent each ngram w within a document d . One of the most popular such approaches is the “*term frequency*” vector:

$$tf_{w,d} = \frac{\text{\#occurences of } w \text{ in } d}{\text{\#ngrams in } d} \quad (2.1)$$

This formula associates a pair of an ngram w and a document d with a score. We can therefore create a $|D_d|$ -by- $|W|$ matrix for d , by converting each ngram w comprising it into its $tf_{w,d}$ vector for this document. The final document representation is generated by summing up the values of its columns⁶. Alternatively, we can also take into account the document frequency of each ngram (i.e., a score df_w , which indicates the number of the documents that w appears in). The inverse document frequency of an ngram is thus defined as:

$$idf_w = \log \frac{|D|}{df_w} \quad (2.2)$$

By considering Eq. 2.1 and 2.2, we can construct what is commonly known as the “*tfidf*” representation as:

⁶There exist a few normalisation approaches on the term frequency matrix, the presentation of which falls out of the scope of this section. For related information, the reader is pointed to [215].

$$tfidf_{w,d} = tf_{w,d} \cdot idf_w \quad (2.3)$$

Finally, every document can be represented again as a $|W|$ -dimensional vector, similarly to the term frequency case.

Lexicon-based features

In domain-dependent NLP tasks the ngram representation is problematic. For example, in sentiment analysis we cannot learn a model based on ngram representations on the sports domain and expect to achieve similar performance in the political domain, since the words and phrases that indicate a positive or negative sentiment vary across these domains [244]. Thus, it is important to establish further feature sets that map every word to a more semantic score, especially for tasks related to opinion mining on user-generated content.

The first such representation we introduce is the lexicon-based document representation. Lexicons are dictionaries that associate a ngram with a certain score or class. For example, a sentiment lexicon will have the word *bad* associated with the “negative” class, or with a score “-1”. Such lexicons have been generated in the past either in a manual fashion or by automatic means. In the first case, experts are manually annotating some words, thus producing some small-scale, but high-quality, word dictionaries. However, such lexicons fail to capture the semantics of the noisy user-generated content, which includes misspellings, abbreviations, smileys/frowns etc. As a result, over the last decade there has been a growing effort on creating such lexicons based on semi-supervised methods over large-scale, user-generated content [159, 163, 160, 164].

One popular approach is based on the concept of pointwise mutual information (PMI) [40]. PMI is a metric that associates a feature w with a class y as follows:

$$PMI(w; y) = \log \frac{p(y|w)}{p(y)}. \quad (2.4)$$

Given a large collection of labelled user-generated documents, we can derive the PMI of each ngram w with each of the classes. In the sentiment analysis context, the classes can be “positive” and “negative”. By computing the PMI of every word and class (pos/neg), we can derive a single score for an ngram as:

$$s_w = PMI(w; pos) - PMI(w; neg). \quad (2.5)$$

Here, a high positive (negative) s_w , implies that w appears much more in positive (negative) documents, thus its presence in a new document might be indicative of the latter’s sentiment.

Using any type of lexicons, feature extraction can be performed in intuitive ways, such as calculating the overall sum of the ngrams’ scores appearing in a document, counting the number of negative ngrams in a document, etc.

Topic similarity

“Topics” are clusters of {word, value} pairs, which group together under the same cluster words that appear often in a similar context. For an introduction to topic modelling approaches [76, 21, 234], the reader is pointed to [136]; for the remainder of this section, we will only present how we can use the output of such methods (i.e., the clusters of {word, value} pairs), since topic modelling is a research field that falls out of the scope of the current Thesis.

Given $|T|$ static topics $\{t_0, \dots, t_{|T|-1}\}$ and a document d comprised of ngrams $\{w_0, \dots, w_{|d|}\}$, we need to find the relevance of d with respect to each of the topics. A commonly used metric that we will employ is the *cosine similarity*:

$$sim(d, t_i) = \cos(\theta) = \frac{d \cdot t_i}{\|d\| \|t_i\|}. \quad (2.6)$$

The value of $sim(d, t_i)$ ranges within the $[-1, 1]$ interval, with higher values indicating higher similarity between d and t_i . Feature extraction is typically performed by computing $sim(d, t_i)$ for every topic t_i and using the resulting $|T|$ -dimensional vector as the topic-based feature set for representing d .

Count-based features

When dealing with noisy user-generated content, there are several other features that might be related to our task. For example, the presence of an all-upper-case word might be indicative of the contained emotion (anger). Albeit not sophisticated, these count- or presence-based features (e.g., presence/number of elongated words, URLs, user mentions, exclamation marks etc.) can offer some supplementary information to the feature sets that were previously presented.

Word Embeddings

All of the aforementioned text representation approaches are based on keyword matching methods. Such methods essentially fail to represent a single word based on its context. Recent advances in generating distributed word representation forms have attracted a lot of interest and have been successfully incorporated in numerous NLP tasks. In these approaches, a single word within a vocabulary of size $|V|$ is represented as a D -dimensional vector (typically, $|D| \ll |V|$) in a latent space, which also accounts for its context.

In this Thesis, we will build and use word embeddings trained primarily on the method proposed by Mikolov et al. [153]. In the so-called “*word2vec*” model, we are given a set of sentences $\{s_1, \dots, s_N\}$, each composed by a sequence of words $\{\{w_{11}, \dots, w_{1M_1}\}, \dots, \{w_{N1}, \dots, w_{NM_N}\}\}$. word2vec introduces a fake task, which comes into two flavours: (a) the *Continuous bag-of-words* (CBOW) model aims at predicting a target word within a subsequence, given the words around it (i.e., “context” words); (b) the *Skip-gram* (SG) model works in an inverse fashion, aiming to predicting the words around a certain word of a subsequence.

Both of these methods (CBOW, SG) are formulated via a one-hidden-layer (with D units) neural network (see next subsection). The input is a one-hot representation of the input word, which is passed through the hidden layer, and the output is the one-hot encoded target word. As we will discuss in the next subsection, neural networks use a loss function which they try to minimise, in order to tune the weights of the input-to-hidden and hidden-to-output matrices

(these weights are the resulting word embeddings). Let us denote the {input, output} vectors as $\{w_O, w_I\}$ and the {input-to-hidden, hidden-to-output} vectors of a word w as $\{v_w, v'_w\}$. A typical function used in neural networks is the softmax function:

$$p(w_O; w_I) = \frac{e^{v'_{w_O} \cdot v_{w_I}}}{\sum_{i=1}^V e^{v'_{w_i} \cdot v_{w_I}}} \quad (2.7)$$

However, updating all vectors in a large vocabulary V makes the training process rather slow. For this reason, other loss functions, such as the Hierarchical Softmax [168], Noise Contrastive Estimation (NCE), [90] or the Negative Sampling proposed by Mikolov et al. [153] are typically used instead. Training takes place over a certain number of epochs and the final word representations are retrieved by the tuned weights of the hidden layer.

The final document representation can be constructed by applying various functions on each of the resulting dimensions of its words (e.g., by taking the average of each dimension of every word within the document). Finally, it should be noted that several other methods for generating such latent representations have been proposed over the latest years [132, 188, 231, 191], such as task-specific word representations [231] or representations that also account for the words' syntax, semantics [191] or document-level information [132].

2.4 Algorithms

In this section we outline the major algorithms that will be used in our experiments throughout this Thesis. Further details (e.g., on parameter tuning) will be provided in the corresponding chapters.

2.4.1 Classification Algorithms

Logistic Regression

Logistic Regression [53] aims at classifying a test instance with an N -dimensional feature vector x as:

$$\hat{y} = f(x; w, b) = \sigma\left(\sum_{i=1}^N (w_i x_i) + b\right) \quad (2.8)$$

where $\sigma(z)$ is the logistic function:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (2.9)$$

We can also plug the bias term b into the first position of the x vector, by also placing the value “1” in the first position of the w vector. Thus, Eq. 2.8 can be re-written as:

$$\hat{y} = f(x; w) = \frac{1}{1 + e^{-w^T x}} \quad (2.10)$$

In a binary classification task, where the instances are labelled as 1 (“positive”) or 0 (“negative”), the probability of an instance to be classified as a “positive” example is provided by Eq. 2.10 (i.e., if $\hat{y} > 0.5$). To assess the model’s effectiveness during training, we can use the log likelihood function:

$$LL(w) = \sum_j y_j \log \sigma(z_j) + (1 - y_j) \log[1 - \sigma(z_j)] \quad (2.11)$$

We can tune the parameters w of our model in Eq. 2.11 via the Gradient Ascent method, which equivalent to the Gradient Descent on $-LL(w)$.

Feed Forward Neural Network

Neural Networks are composed by several layers, which in turn are composed by several units (“neurons”). A neuron performs the following operation on some input x :

$$f(x; w, b) = f\left(\sum_{i=1}^N (w_i x_i) + b\right), \quad (2.12)$$

where f is some “activation” function, such as the Rectified Linear Unit (ReLU), \tanh , or the sigmoid function σ which was also used in Logistic Regression. Typically, each layer is formed by hundreds of neurons, each associated with its own weight matrix W and bias terms b . The outputs of all of the neurons’

functions f within a layer are then used as an input for the neurons in the next layer. This way, we can stack up several layers, each with a different number of neurons and activation functions. In the last layer, the final (binary) classification takes place through a sigmoid activation function.

Parameter tuning takes place through backpropagation. There are several optimisers that can be used to tune the neural network’s weights based on some loss function. In this Thesis, we will use the Adam optimiser [120], which is an extension to the Stochastic Gradient Descent, to minimise the binary cross entropy function:

$$L(w) = -(y \log(p) + (1 - y) \log(1 - p)), \quad (2.13)$$

where p denotes the probability of the instance to be assigned to a certain class by our model and y its true label (0 or 1). Instead of calculating the gradient and updating the neural network’s weights on every example, we can instead form “mini-batches” composed by several instances and compute the gradient against each mini-batch to speed up the training process, which terminates after a certain number of epochs. Common regularisation techniques include the L_2 regularisation and the incorporation of the “Dropout” layer (i.e., deactivating a pre-defined percentage of neurons in a certain layer during training).

Random Forest

Decision trees are a class of machine learning algorithms that aims to learn a function mapping the features x of a training set $\{x, y\}$ to their target label y (or score, in a regression setting) on the basis of IF-ELSE-AND operations. In particular, a decision tree is formed by interior nodes, each of which corresponds to a particular feature x_i and its associated values upon which the split at this node will occur, and leaf nodes, each of which corresponds to a certain target score. The metrics upon which the nodes are selected vary depending on the implementation; typical such metrics include the Gini Impurity and the Infor-

mation Gain. After training, the prediction of the label y^* of a test instance x^* takes place by browsing through the path of the formed tree: at every node met along the path leading from the root to a leaf node, the respective feature value x_i^* is compared with the values of this feature that are used for splitting this node, leading to the next node, where the same comparison is performed, until the end. Despite providing desirable explanatory predictions in the form of IF-ELSE-AND operations, Decision trees are vulnerable to overfitting and appropriate pruning approaches are often introduced, in a regularisation attempt. For an overview of decision trees, the reader is pointed to [139].

Random Forest [95, 32] is an ensemble model, which employs numerous decision trees as its building blocks, aiming to tackle their overfitting issue. Learning takes place by training K different decision trees on some randomly sampled instances of the same size N , leading to K different models:

$$\hat{y}_k = f_k(x_k).$$

As opposed to traditional Decision Trees algorithms, during the training process of a Random Forest model, a random sample of the available features is selected at each split of a tree. This results into generating an ensemble model which is not dependent on a few highly predictive features based on the observations of the training set, and thus it helps in avoiding overfitting. At the final step, the prediction of a test instance is made based on the majority vote over the K decision trees.

Support Vector Machine

Support Vector Machines (SVMs) [52, 29] are powerful classification algorithms that have been employed in numerous NLP tasks over the latest decades. In this Thesis, we employ SVMs for multiple tasks and propose different kernel-based approaches for temporal modelling of asynchronous information, by employing SVMs as the core model of our proposed solutions.

Let $\{x_j, y_j\}$ be a set of instances, with $y_j \in \{-1, 1\}$ and $x_j \in R^N$. Recall that Logistic Regression classifies an instance with features x as “positive”, iff $\sigma(z) > 0.5$, with $z = \sum_{i=1}^N (w_i x_i) + b$. SVMs aim instead at finding a hyperplane that separates the data in a way such that its distance from the nearest data points of both classes is maximised. This *separating hyperplane* is defined by the line:

$$\sum_{i=1}^N (w_i x_i) + b = w^T x + b = 0,$$

where b indicates the intercept term. The *geometric margin* γ to the hyperplane is defined as:

$$\gamma = \min_j \left(y_j \left(\frac{w^T x_j + b}{\|w\|} \right) \right) \quad (2.14)$$

SVMs try to maximise the geometric margin γ in Eq. 2.14. This is equivalent to solving:

$$\min_{\gamma, w, b} \left(\frac{\|w\|^2}{2} \right), \quad \text{s.t. } y_j (w^T x_j + b) \geq 1, \forall j. \quad (2.15)$$

We can also express Eq. 2.15 in its dual form, by introducing the Lagrange multipliers α , as follows:

$$\begin{aligned} \max_{\alpha} f(\alpha) &= \sum_j \alpha_j - \frac{1}{2} \sum_j \sum_{j'} y_j y_{j'} \alpha_j \alpha_{j'} \langle x_j, x_{j'} \rangle \\ \text{s.t. } \alpha_j &\geq 0, \forall j \\ \sum_j \alpha_j y_j &= 0. \end{aligned} \quad (2.16)$$

Eqs. 2.15-2.16 assume a linearly separable case between the two classes. Under a real-world setting though, this is rarely the case. Thus, often the concept of *regularisation* is introduced, transforming the respective formulas

into:

$$\begin{aligned}
 & \min_{\gamma, w, b} \left(\frac{\|w\|^2}{2} + C \sum_j \xi_j \right) \\
 & \text{s.t. } y_j(w^T x_j + b) \geq 1 - \xi_j, \forall j \\
 & \xi_j \geq 0, \forall j,
 \end{aligned} \tag{2.17}$$

where ξ_j are slack variables and C is a soft-margin parameter that we tune during development, and

$$\begin{aligned}
 \max_{\alpha} f(\alpha) &= \sum_j \alpha_j - \frac{1}{2} \sum_j \sum_{j'} y_j y_{j'} \alpha_j \alpha_{j'} \langle x_j, x_{j'} \rangle \\
 \text{s.t. } C &\geq \alpha_j \geq 0, \forall j \\
 \sum_j \alpha_j y_j &= 0.
 \end{aligned} \tag{2.18}$$

For calculating the optimal parameters α_j , the Sequential Minimal Optimisation (SMO) algorithm is often employed [192]. The final prediction is then performed as:

$$\hat{y} = f(x^*) = \text{sign} \left(\left(\sum_j \alpha_j y_j x_j \right)^T x^* + b \right) \tag{2.19}$$

Kernels. *Kernels* can be incorporated into SVMs in order to capture non-linear relations in the input space. Note that SVMs can be expressed entirely in terms of inner products $\langle x, x' \rangle$. Hence, we can define a function, called the *kernel*, $k(x, x') = \langle \phi(x), \phi(x') \rangle$, using some feature mapping ϕ over the original input space. This way, we can transform the original input into a higher-dimensional feature space. Importantly, we can calculate straight away the “kernel function” $k(x, x')$ without explicitly calculating $\phi(x)$ and $\phi(x')$. By applying the same kernel function to all pairs $\{x, x'\}$ of our input space, we can compute the “kernel matrix” K as:

$$K = k(x, x'), \forall x, x'. \quad (2.20)$$

In this Thesis, we will primarily use the radial basis function (RBF) as our kernel function, which is defined as:

$$k(x, x') = e^{-\gamma \|x - x'\|^2}, \quad (2.21)$$

where γ is a parameter that we tune during development. A small γ implies a function with large variance – and vice versa. The final RBF kernel can also be viewed as a instance-wise similarity matrix, with high (low) values, indicating more (less) similar instances.

Using a kernel-based SVM can be done by replacing the inner product $\langle x_j, x_{j'} \rangle$ in Eq. 2.18 with $K(x_j, x_{j'})$. The final prediction is performed as follows:

$$\hat{y} = f(x^*) = \text{sign} \left(\sum_j \alpha_j k(x_j, x^*) + b \right). \quad (2.22)$$

Importantly, we can form a valid kernel K by summing different kernels K_0, K_1, \dots, K_N , whereas multiplying two kernels also yields a valid kernel. Though we will not provide further formal details on kernels and kernel-based methods here, the reader is pointed to the works by Hofmann et al. [97] for this purpose.

Multiple Kernel Learning

Often, we need to model different aspects of our problem, each via a different kernel. For example, for a micro-level binary user classification task, we might be interested in predicting a target class \hat{y} , using data derived from the user's tweets, images, URLs, his/her followers, etc. One approach to deal with this task would be to model each of these modalities via a RBF kernel and then sum up the resulting kernel matrices, to derive a final kernel K . However, this would imply an assumption that each of these modalities are equally important for our task.

Multiple Kernel Learning (MKL) methods aim at building a model composed by different kernels, by learning both of the model's parameters and the weights of the individual kernels at the same time. In this Thesis, we employ an SVM-based MKL approach proposed by Sonnenburg et al. [227]. In specific, in a binary classification setting, we make a prediction based on the following:

$$\hat{y} = f(x^*) = \text{sign}\left(\sum_j \alpha_j \sum_s w_s K_s(x_j, x^*) + b\right),$$

where K_s represent the kernels of the different modalities. Note that the difference compared to Eq. 2.22 is the replacement of $k(x_j, x^*)$ with the linearly weighted summation of the kernels, denoted by $\sum_s w_s K_s(x_j, x^*)$. The parameters α_j , the bias term b and the kernel weights w_s are estimated by minimising the expression:

$$\min \quad \gamma - \sum_j \alpha_j \tag{2.23}$$

$$\text{w.r.t.} \quad \gamma \in R, \alpha \in R_+^{|J|}$$

$$\text{s.t.} \quad 0 \leq \alpha_j \leq C \quad \forall j, \quad \sum_j \alpha_j y_j = 0$$

$$\frac{1}{2} \sum_{j,j'} \alpha_j \alpha_{j'} y_j y_{j'} K_s(\mathbf{x}_j, \mathbf{x}_{j'}) \leq \gamma \quad \forall s. \tag{2.24}$$

Importantly, by taking advantage of the kernel properties (e.g., summation, multiplication), we can model different modalities and combine them in a weighted scheme that also accounts for their contribution with respect to the prediction task. This gives us the flexibility to add kernels that are even unrelated to our goal, without the need to decide a-priori which of them are important for our prediction task, as we will show in Chapters 6 and 7 of this Thesis.

2.4.2 Regression Algorithms

Linear Regression

In *Linear Regression*, we aim at predicting the target score \hat{y} of an instance with an N -dimensional feature vector x as a linear combination of some weights w :

$$\hat{y} = f(x; w, b) = \sum_{i=1}^N (w_i x_i) + b \quad (2.25)$$

We can also plug the bias term b into the weights vector w by also placing the value 1 at the first position of the feature vector x and shifting the rest of its values. This way, we can re-write 2.25 as:

$$\hat{y} = f(x; w) = \sum_{i=0}^N w_i x_i \quad (2.26)$$

The parameters w_i are estimated based on the observations on the training set. In particular, during training, we try to minimise a loss function L , which is usually the sum of least squares, given by the expression:

$$[w^*] = \underset{w}{\operatorname{argmin}}(L(w)) = \underset{w}{\operatorname{argmin}} \left(\sum_j \left(y_j - \sum_{i=0}^N w_i x_{ji} \right)^2 \right) \quad (2.27)$$

Eq. 2.27 has a closed form solution; however, this is often hard to compute and it is not applicable to an online learning setting. Thus, optimisation methods, such as Gradient Descent, are often preferred. In particular, in Gradient Descent, we update our model's weights based on some learning rate η as:

$$w^* = w - \eta \frac{\partial L}{\partial w} \quad (2.28)$$

This iterative process stops based on some criterion (e.g., a pre-defined maximum number of iterations, a certain threshold we aim to achieve for our loss function score L , etc.). After tuning w , we can use them under Eq. 2.26 to predict the scores of a test instance.

Least Absolute Shrinkage and Selection Operator (LASSO)

Linear Regression is vulnerable to overfitting, especially when the size of the feature vector x is large. To accommodate that, we introduce the concept of *regularisation* under the *Least Absolute Shrinkage and Selection Operator (LASSO)* model. In particular, we add another term to Eq. 2.27:

$$[w^*, b^*] = \underset{w, b}{\operatorname{argmin}}(L(w, b)) = \underset{w, b}{\operatorname{argmin}} \left(\sum_j \left(y_j - b - \sum_{i=1}^N w_i x_{ji} \right)^2 + \lambda \sum_{i=1}^N |w_i| \right), \quad (2.29)$$

where λ is a parameter that controls the balance between the level of regularisation and our previously defined loss function, and $\sum_{i=1}^N |w_i|$ is the L_1 -norm of the weight vector w . This regularisation term essentially results in reducing the magnitude of the elements in w ⁷, thus helping in avoiding overfitting to some feature x_i (with a corresponding high value w_i), which is often a problem in NLP tasks, owed to the high dimensionality of the feature vector x .

Random Forest

Random Forests for regression operate in a similar manner and have the same properties with their corresponding classification models. The only major difference between the two is that, in a regression setting, the final prediction takes place by taking the average of the K predictors (i.e., instead of the majority vote that is used in the classification setting):

$$\hat{y} = \frac{1}{K} \sum_{k=0}^{K-1} \hat{y}_k. \quad (2.30)$$

Support Vector Regression Machines

Support vector regression machines (“SVRs”) [66] are an extension to SVMs, dealing with regression tasks. Compared to SVMs, SVRs aim at solving the following optimisation task:

⁷Note that the bias term b is not regularised.

$$\begin{aligned}
 & \min_{\gamma, w, b} \left(\frac{\|w\|^2}{2} \right) \\
 & \text{s.t. } y_j - \langle w, x_j \rangle - b \leq \epsilon, \text{ and} \\
 & \quad \langle w, x_j \rangle + b - y_j \leq \epsilon, \forall j.
 \end{aligned} \tag{2.31}$$

Similarly to the case of SVMs, we can introduce a constant C to allow for some errors, while introducing slack variables $\xi_j, \xi_j^* \geq 0$, as follows:

$$\begin{aligned}
 & \min \left(\frac{\|w\|^2}{2} + C \sum_j (\xi_j + \xi_j^*) \right) \\
 & \text{s.t. } y_j - \langle w, x_j \rangle - b \leq \epsilon + \xi_j, \text{ and} \\
 & \quad \langle w, x_j \rangle + b - y_j \leq \epsilon + \xi_j^*, \forall j.
 \end{aligned} \tag{2.32}$$

The dual form is provided by maximising the following:

$$\begin{aligned}
 & -\frac{1}{2} \sum_{j, j'} (\alpha_j - \alpha_j^*)(\alpha_{j'} - \alpha_{j'}^*) \langle x_j, x_{j'} \rangle - \epsilon \sum_j (\alpha_j + \alpha_j^*) + \sum_j y_j (\alpha_j - \alpha_j^*) \\
 & \text{s.t. } \sum_j (\alpha_j + \alpha_j^*) = 0, \quad C \geq \alpha_j \geq 0, \quad C \geq \alpha_j^* \geq 0, \forall j,
 \end{aligned} \tag{2.33}$$

where α_j, α_j^* are Lagrange multipliers. The prediction of the SVR takes place based on:

$$\hat{y} = f(x^*) = \sum_j (\alpha_j - \alpha_j^*) \langle x_j, x_{j'} \rangle + b. \tag{2.34}$$

Again, as in the case of SVMs, the algorithm is described by dot products of the input space. Thus, we can incorporate kernels, by replacing $\langle x_j, x_{j'} \rangle$ with $K(x, x^*)$ in Eq. 2.34. In the same sense, we can also incorporate the *MKL* approach [227] in a regression setup, similarly to its equivalent classification setting. For more details on SVRs, the reader is pointed to the work by Smola and Schölkopf [225].

2.5 Evaluation

To assess a model’s ability to learn an effective mapping via a function $\hat{y} = f(x)$, appropriate evaluation metrics need to be introduced. Before presenting the metrics that we have used in this Thesis, we outline the two major types of evaluation that are most commonly used in past work.

2.5.1 Validation Approaches

Given some instances $\{x, y\}$, we are asked to learn a function f that learns the association between x and y , so that it can provide accurate estimates of a new instance x^* . To achieve that under a real-world setting, we first need to assess the model’s ability to learn the observed examples (i.e., by minimizing some error over the observed examples) and tune its parameters (e.g., the variance of an RBF kernel or the regularisation strength in an SVM). In other words, we need to use our instances in a pseudo-real-world setting, assuming that some of them are the ones that we have in hand (training set) and the rest are the unobserved ones (test set), upon which we test our model’s accuracy. There are two ways of separating our data, and thus test our model’s effectiveness, which are briefly discussed here.

Train/Validation/Test Split. Given some instances $\{\{x_0, y_0\}, \dots, \{x_N, y_N\}\}$, we can split them in three sets: (a) the *training set* (typically, consisting of 50%–80% of the instances) is used to learn a model on; (b) the *validation set* (typically, 10%–20% of the instances) is optionally used in order to tune the parameters of our model and/or get an estimate of its performance on previously unobserved examples; (c) the *test set* (typically 10%–30%) is used only for testing our pre-trained and pre-tuned model on and measuring its performance. Since we know the labels of the test set instances, the evaluation is typically performed on this set.

Cross-validation. The problem with the train/validation/test set is that when the number of our instances $|N|$ is small, then we might end up training and validating our model on a very few instances, thus not ensuring (even empirically) similar performance over new examples. To accommodate this, we can instead use a *K-fold cross validation* (typically, $K = 10$), where we first place all of our instances into K equasized bins and then we use $(K - 1)$ bins as our training/validation sets and the remaining one for test purposes. By repeating this process K times, we get an overall performance of our model (e.g., by averaging its performance over the K runs), while all of our instances have served for testing purposes.

2.5.2 Evaluation Metrics

In this subsection, we assume that we have learned a function f based on some training examples and we have applied it in a test set of new instances with true labels y , making predictions \hat{y} . Table 2.1 provides the definitions of the evaluation metrics used throughout this Thesis, which are discussed in the next two paragraphs.

Regression Metrics. In regression tasks, both y and \hat{y} are numerical. Thus, we need to introduce some measurements of *error* of our models, with lowest values indicating a better model. The most intuitive such metric is the mean absolute error (*MAE*), measuring the average absolute difference of each pair of $\{y_i, \hat{y}_i\}$. Similarly defined error metrics are the mean squared error (*MSE*) and the root mean squared error (*RMSE*), which are used to penalise more the more inaccurate predictions. A drawback of such error metrics is that they fail to measure how much of the variance of our target y can be explained by our model. The coefficient of determination (R^2) accounts for this fact, aiming to measure how better our model is compared to the average predictor (that is, compared to the naïve model that always predicts the average of the test set instances \bar{y}). This metric is important, especially when dealing with micro-

level tasks, where our goal is to predict a time-varying user index, using his/her average index score as our naïve baseline.

Classification Metrics. In classification tasks, both y and \hat{y} are classes. For every class c in our test set, we can define the following basic concepts:

- **True positive** (tp) indicates the number of instances that have been correctly predicted as belonging to class c .
- **True negative** (tn) indicates the number of instances in the test set that have been correctly predicted as not belonging to class c .
- **False positive** (fp) indicates the number of instances that have been wrongly predicted as belonging to class c .
- **False negative** (fn) the number of instances that have been wrongly predicted as not belonging to class c .

Based on these concepts, we can define our classification metrics, as listed in Table 2.1. *Accuracy* measures the percentage of correctly classified instances. *Precision* is the ratio of correctly classified instances for a class c to the overall number of instances that were classified as belonging to c . *Recall* (or “sensitivity”) is similarly defined by the ration of correctly classified instances for a class c to the total number of instances that belong to c . Finally, the *F1-score* is a commonly used metric that integrates the concepts of both the precision and the recall. These metrics are computed either in an average across-all-classes (macro-average) or in a micro-averaging fashion. For the most part of this Thesis, we will be presenting results based on the macro-average F1-score (or simply, “F-score”), which is more challenging under a real-world setting, due to the imbalanced nature of most datasets we work on, while acting as a harmonic means between precision and recall.

Classification		Regression	
Accuracy	$(tp + tn)/(tp + fp + tn + fn)$	MAE	$(\sum_i y_i - \hat{y}_i)/N$
Precision	$tp/(tp + fp)$	MSE	$(\sum_i (y_i - \hat{y}_i)^2)/N$
Recall	$tp/(tp + fn)$	RMSE	$\sqrt{(\sum_i (y_i - \hat{y}_i)^2)/N}$
F-score	$2 \cdot precision \cdot recall / (precision + recall)$	R²	$1 - (\sum_i (y_i - \hat{y})^2) / (\sum_i (y_i - \bar{y})^2)$

Table 2.1: Summary of the evaluation metrics used in this Thesis.

2.6 Summary and Conclusion

In this chapter, we have introduced the research process followed in this Thesis, as well as some of the major concepts that are needed for the comprehension of the next parts. In the next chapter, we provide an overview of past work related to our tasks at the macro- and micro-level, which form the major part of this Thesis (Parts II and III).

CHAPTER 3

Related Work

In the current chapter, we outline the related work for each task that is tackled in this Thesis. We begin with a high-level introduction around monitoring user behaviour through social media and smart devices. Then, we link each research question (RQ) that was presented in Chapter 1, with the work that has been conducted in order to address it in the past. We present the open questions and the motivation of our approaches that will follow in the next chapters.

3.1 Monitoring User Behaviour through Social Media and Smart Devices: Overview

Over the latest decade, there is an increasing research interest in the high-level field of monitoring user behaviour through social media and smart devices. Such work aims to extract features out of the noisy user-generated content and learn models that map it to real-world indices, such as political [247, 216, 243, 34, 35, 113], health-related [193, 55, 124, 182, 67, 261, 127, 260] or socioeconomic indices [26, 206, 202, 199, 138, 13, 176]. As discussed in Chapter 1, these tasks are either performed on the macro- (population-wide) or on the micro-level (user-based). In this Thesis, we present our approach in different tasks at both levels. In the current section, we outline the related work in each of these tasks: *(a)* first, we provide a generic background on sentiment analysis on user-generated content and we present the related work on generating textual resources for tasks related to opinion mining – a problem which we will tackle in the next chapter; *(b)* next, we present approaches on both macro- and micro-level, related to the task of nowcasting political-related indices, which will be presented in Chapters 5 and 6, respectively; *(c)* finally, we present a thorough literature review on past approaches on nowcasting mental health indices – a task which we will tackle on the final chapters of this Thesis – as well as ethical considerations that need to be addressed when mining user-generated content.

3.2 Sentiment Analysis

Sentiment analysis (SA) [181] is the task of classifying a piece of text with respect to its sentiment, which is usually defined as “positive”, “negative” or “neutral”. Depending on the level of the analysis, the task can be refined (among others) as “message-level”, “topic-based” [211], “target-specific” [254, 143, 165], “aspect-based” [195, 194] or “phrase-level” SA. “Message-level” SA aims at classifying a whole document with respect to its class; “topic-based” SA is the

task of classifying a piece of text with respect to a particular topic; the goal of of “aspect-based” SA is to classify the sentiment conveyed in a message with respect to a particular aspect (e.g., the sentiment towards the communication skills of the employers in a hotel); “phrase-level” SA classifies particular words or phrases within a document (e.g., the word “happy” or the phrase “I am feeling sad”); finally, “target-specific” SA aims at classifying the sentiment expressed towards different entities within a document. In this Thesis, we will focus on the message-level SA, which is the most popular sub-task in the field of sentiment analysis, aiming to develop resources that can be used for feature extraction in order to lead to significant improvements in performance, compared to the more traditional ngram features. However, an emerging sub-field of SA that has also received a lot of research interest recently is that of target-specific SA; for an overview and evaluation of existing approaches on this task, the reader is pointed to the work by Moore and Rayson [165].

Early approaches on message-level sentiment analysis¹ on Twitter relied primarily on ngram features [84, 178, 20]. However, ngrams have been empirically proven to be insufficient when the trained model is applied in a different domain [208, 8, 244]. To mitigate this effect, features derived from sentiment lexicons have been incorporated in the task [258, 8, 16, 105, 244], also providing better performance for the in-domain SA task, an important level of coverage [86] and rendering high precision rates [117]. Moreover, lexicons can better handle negation and intensification [230], as well as improve the performance of opinion retrieval systems [106]. Prior to the era of dealing with the noisy nature of micro-blogging platforms, several lexicons have been developed, mapping a single word or phrase to its sentiment or emotion [100, 258, 71, 56, 230]. As discussed in the previous chapter, several lexicons have also been developed to cope with this noisy nature of user-generated content, leveraging large-scale textual resources available through social media and associating each ngram with a sentiment score (e.g., using the PMI metric) [160, 259]. The incorporation of

¹For simplicity, we will refer to the task of “message-level sentiment analysis” as “sentiment analysis”.

the lexicon-based features based on the latter has offered strong performance gains in the task of sentiment analysis on micro-posts [160, 259].

Over the last five years, more emphasis is paid on employing latent feature representations [153, 152], most often through deep learning approaches (i.e., neural network models that incorporate several layers in their architecture) [114, 119]. Such approaches use dense word representations in their input and learn sentence-level representations, by fine-tuning their parameters depending on the task. In the 2017 SemEval competition, the top-performing teams [17, 41] for SA in the English language employed deep learning approaches; however, a Complement Naïve Bayes [69] and an SVM [102] classifier using combinations of linguistic features achieved the state-of-the-art for the same task in the Arabic language, outperforming a deep learning approach [87]. Furthermore, the performance of feature-rich kernel-based methods [102] is comparable to state-of-the-art deep learning methods [17, 41], even in the English language, in which the latter have been the dominant approach over the past few years. Finally, tuning the parameters of a deep learning model requires a lot of data to train on in most cases, which might not be available in a real-world setting.

Given the aforementioned facts, the task of generating resources that can be effectively used in SA tasks becomes of great importance, since such resources can help in opinion mining on the large scale. In this Thesis, we tackle the problem of generating such rich resources in the Greek language (see Chapter 4). The Greek language is of particular interest owed to its highly inflected nature and the need for online monitoring of real-world events occurring in Greece, due to the economic crisis. In the next subsection, we describe past work on generating such lexical resources on non-English languages and outline the challenges with respect to generating such resources for the Greek language.

3.2.1 Generating Sentiment Resources (preliminary analysis)

Past work on generating lexical resources in non-English languages has primarily relied on translations of English-based sentiment lexicons and mappings of WordNet synsets, to transfer the polarised words from English to the target language [106, 56, 10, 189], while common tools for expansion methods of the generated lexicon include Part-Of-Speech (POS) taggers [250] and syntactic rules [106]. In particular, Das and Bandyopadhyay [56] used the Subjectivity Word List [258] and leveraged WordNet synsets to create a lexicon for the Indian languages, which was further expanded using a corpus-based approach. In [250], a similar approach was used for generating an initial lexicon for the Indonesian language, which was expanded using different methods, such as finding words in common patterns of three-grams with positive/negative words in a corpus. Perez-Rosas et al. [189] showed that bridging the language gap between English and Spanish languages using the multilingual sense-level aligned WordNet structure allows to generate a high accuracy polarity lexicon. Other approaches include a PageRank-like algorithm that was used in [107] for creating a lexicon in Dutch based on the relations of the WordNet synsets; synonym and antonym relations have been used for expanding a lexicon for Hindi by Arora et al. [10], while the use of word affixes has also been exploited by Mohammad et al. [159]. With respect to generating resources specifically for the Greek language, Palogiannidi et al. [179] translated English words from the ANEW lexicon [31] and manually annotated them with respect to their valence, arousal and dominance. Other works on sentiment-related tasks in the Greek language have not created and comparatively evaluated linguistic resources for such tasks [1, 226].

As there do not exist any reliable syntactic parsing and POS tagging tools for the Greek language, making use of such resources [250, 106] is not possible in our case, while language-dependent word-level rules [159] cannot generalise; also, translation techniques and WordNet synset mapping [106, 56, 10, 189] are

risky and ineffective when dealing with noisy content. Furthermore, none of the above works has evaluated the generalisation capabilities of the generated resources with respect to different tasks from different domains. Other approaches, such as translating the documents from the target language into English, have shown surprising improvements in performance of sentiment analysis models [162], but those are expensive and cannot be applied with high confidence in a highly inflected language, such as Greek. Last but not least, to the best of our knowledge, the only work that has focused on the Greek language, by Palogianidi et al. [179], created a lexicon of words with respect to their valence, arousal and dominance and not to their sentiment or emotional orientation. While such emotional dimensions of a word might indeed be helpful in a sentiment classification task, they are not as explicit as the standard subjectivity and polarity labels of the words for the sentiment analysis task.

In this Thesis, we refrain from performing word translation. Instead, we manually annotate 2.2K words in Greek (32K in their expanded forms) with respect to their sentiment and emotion and we follow past approaches on the English language to *(a)* expand our original lexicon and *(b)* account for the noisy user-generated content of social media. We also generate dense word representations in Greek, leveraging large-scale streams of textual data, which we incorporate in our analysis. Through a thorough experimentation over multiple tasks, we demonstrate the effectiveness of our resources, achieving state-of-the-art results in SA in the Greek language and offering further improvements in related tasks over the ngram baseline. To the best of our knowledge, our lexicons and word representations are the first publicly available, large-scale and systematically evaluated resources for the Greek language.

3.3 Social Media and Elections (RQ1, RQ2)

The large volumes of user-generated data produced in online social networks have attracted a lot of research interest in mining these rich resources for various

purposes. Owing to the opinionated nature of a large proportion of the comments produced in social media, a growing body of work focuses on mining such data streams for tasks related to the political domain, the most popular of which being the task of predicting the election results at the macro-level. In the current section, we outline past work in this domain, both at the macro- as well as at the micro-level, which we will tackle on Chapters 5 and 6, respectively.

3.3.1 Social Media and Elections at the Macro-level (RQ1)

The task of mining social media to predict the election results has received much attention over the latest decade, with approaches and results varying. Such work aims at extracting features from online social media that are related to the political domain, and correlate with or train models that predict the final election results. Examples of such features can include the number of times a political party’s name appears in social media over a certain time interval preceding the elections, the average sentiment expressed towards a political party, etc. Past work has studied the task either in a feature aggregate [247, 216, 224, 63] or in a temporally-sensitive setting [173, 19, 126, 37].

Early work by Tumasjan et al. [247] showed that simply counting the number of times a German political party’s name appears on Twitter before the 2009 German federal elections can provide a rather accurate estimate of the party’s voting share. Similar findings have been reported by DiGrazia et al. [63] on 795 electoral races over the 2010 and 2012 US congressional elections. However, the approach by [247] has been unsuccessfully applied to other electoral cases [150, 81, 39, 78, 224, 216, 79], demonstrating that such naïve approaches cannot generalise. Furthermore, work by Jungherr et al. [111] also showed that the reported results on the case study of the 2009 German federal elections are based on several constraints. This led into a series of works, primarily by Daniel Gayo-Avello, Takis Metaxas and Eni Mustafaraj [150, 81, 39, 78, 79], summarising the major drawbacks of past work owed to bias in the processing of the social media data with the goal of “matching” the extracted features to

the actual results. The authors emphasise, among others, the need for providing forecasts *before* the end of the elections, in order to evaluate a prediction model for the task in an unbiased way.

A different strand of past work has incorporated the use of opinion polls in their analysis in a time-sensitive manner. Early work by O'Connor et al. [173] studied the correlation of Twitter-based features and opinion polls; however, the evaluation of the time-series models was based on the reports by the opinion polls, which are noisy, by nature. Work by Ceron et al. [37] studied the temporal correlation between Twitter-based features and opinion polls reporting on the popularity of Italian party leaders, as well as the correlation between such features and the voting shares of the leaders and parties competing for the French presidential and legislative elections respectively; however, their analysis was performed post-hoc to the election results. Bermingham and Smeaton [19] used volume- and sentiment-based features extracted from Twitter for the case of the 2011 Irish general elections. Forming a regression task, they trained a model on opinion polls, achieving 3.67% MAE for the five major political parties, if compared against the opinion polls. However, the MAE increases to 5.85%, if we compare it against the actual election results. In a similar fashion, Lampos et al. [126] developed a bilinear model for predicting the opinion polls in the UK and Austria; however, similarly to O'Connor et al. [173], their model has not been validated against the actual election results.

The list of the published work in the domain is non-exhaustive. For the most up-to-date systematic review on the domain, the reader is referred to the work by Jungherr [110]. Daniel Gayo-Avello also provides a more complete overview of the role of social media in the political domain [80]. Here, we outline the major drawbacks of past approaches, which we aim to tackle in our work in Chapter 5:

- A prediction model of the election results, should be made *before* the end of the elections, in order to achieve unbiased estimates. To the best of our knowledge, all of the past work in the domain have published the estimates

of the model after the elections were held. Daniel Gayo-Avello [80] points that up to 2016, there were only two works that actually predicted the election results: ours and work by Burnap et al. [34]; however, the latter was published after our work that will be presented in Chapter 5.

- Applying the same model in multiple electoral cases with different characteristics and language used is essential, in order to test the ability of the model to generalise in multiple cases under a real-world setting.
- Despite the fact that there is a large body that models the Twitter-based features in a temporal fashion, the model evaluation in most cases is performed against opinion polls, which are noisy. Modelling user-generated content in a temporal fashion and comparing the model performance against feature aggregate approaches should, in principle, lead to improvements in performance, consistently across different electoral races.
- Naïve (e.g., count-based) approaches cannot be considered as competitive baselines. In order to provide insights on the appropriateness of a model for this task under a real-world setting, its performance should also be compared against traditional state-of-the-art approaches (i.e., opinion polls).

Despite addressing these issues in the corresponding chapter, it should be noted that a key challenge in this task lies in its evaluation part. Using opinion polls as our target variable can be misleading, owed to their noisy nature. Thus, the model’s performance can only be appropriately assessed if we compare its estimates against the final election results. However, there are only a few political parties that compete against each other in an electoral race; hence, such an evaluation is relatively weak, even if the model estimates are provided before the end of the elections and the model is tested under multiple electoral cases. Consequently, there is also the need to adjust the task into a micro-level classification problem, where our aim will be to classify the stance of a single

user – a task which is more challenging under a sudden setting, as presented next.

3.3.2 Social Media and Elections at the Micro-level (RQ2)

A smaller body of work has focused on nowcasting the voting intention of social media users in the micro-level, primarily owed to the lack of well-established ground-truth, unless the social media users are manually annotated with respect to their voting intention. Within the political domain, most part of past work has focused on the task of separating users based on their political leaning [207, 47, 186, 3, 30, 44, 251, 200]. Most of this work relies on extracting features from the user’s tweets and training a machine learning model that can separate them either under a classification [207, 47, 186, 3, 30, 44] or a regression setup [200].

Early work by Rao et al. [207] used 1,000 manually annotated Twitter users with respect to their political leaning in the US (Democrats vs Republicans). The authors extracted various textual features and trained SVM classifiers for the binary task of predicting their political leaning, achieving 83% accuracy in their best-performing setting. Follow up work has also considered extracting features from the users’ social interactions for the same task [186, 3, 47], leading to improvements in performance. In particular, Al Zamal et al. [3] showed that the performance of an SVM classifier improves by more than 4%, when features are extracted from the user as well as his/her friends’ tweets; Pennacchiotti and Popescu [186] tested a Gradient Boosted Decision Trees model using various linguistic, social network-based and Twitter-specific features from 10K users achieving 87.75% accuracy, demonstrating that social network features are the most predictive for the task; similarly, Conover et al. [47] employed an SVM model for classifying 1,000 manually annotated users, showcasing an improvement of 15% in accuracy when the features used by the classifier are extracted based on the re-tweeting activity of the users rather than based on the content of their tweets. However, Cohen and Ruths [44] demonstrated that

the promising results achieved in these studies are vulnerable to bias, since they are trying to classify users that are clearly stating their political leaning in their profile (that is, they are “vocal” users). In their work, they showcased that the performance is highly affected when the model is trained and applied on modest – rather than vocal – users, with the accuracy dropping from 84% down to 68%. To accommodate this, a novel approach was proposed by Preotiu-Pietro et al. [200], asking Twitter users to self-declare their political ideology in a seven-point scale, which also accounts for non-binary labelling. In line with [44], the authors showcased that the task of predicting these users’ political ideology is much more difficult than predicting the political ideology of vocal users.

None of the aforementioned works have actually studied the task of now-casting the voting intention of social media users under a real-world setting, over time. This task is especially more challenging under a sudden electoral case (i.e., referendum), when the users are asked to decide upon their stance suddenly and on a short time interval. Within the referendum domain, Fang et al. [72] classified Twitter users as “Yes” or “No” voters in the Scottish Independence Referendum. They labelled the users based on the presence of polarised hashtags in their tweets and incorporated topic models based as their features. Similarly to the previous works, their approach neither performs user stance classification over time nor incorporates temporal modelling, whereas their ground-truth is acquired based on distant supervision, which might result into a low-quality test set for evaluation. Stewart et al. [228] studied the use of language in the Catalan Independence Referendum; however, they did not attempt to perform any user classification task. In a closely linked work, Zubiaga et al. [263] worked on three different independence movements, aiming to classify users with respect to their stance. They employed different classification algorithms trained on textual, network-based and activity-based features, achieving the highest accuracy when the network features based on the user “following” relationships on Twitter are used as an input, in all three studied cases. However, their ground-truth is acquired with keyword-matching methods

based on the profile information provided by the users, which is problematic as depicted by [44], whereas their modelling does not account for the temporal modelling component of the task.

Past work on studying the temporal dynamics on social media during times of crisis [27] has not incorporated a temporal modelling component for the task of inferring a user’s stance. While a user’s stance in our modelling is considered to be a static index, as we will show in Chapter 6, *the temporal modelling of user-generated content* is particularly important for capturing this static index under a real-time setting: as real-world events take place in different points in time, users react towards them in a similar timely ordered manner; capturing such user similarities across time (and not in an aggregate manner) is essential in order to build real-world monitors of user behaviour. Other studies have focused on measuring the polarisation of the network structure during times of crisis [166]; however, they have not worked on the task of inferring users’ voting intention neither. Finally, a few works around the Greek Referendum, which is our case study, have focused primarily on analysing the content shared during its short duration [9, 151] and have not studied the task of inferring a user’s voting intention.

To this end, we identify the following gaps in related work, which we aim at covering in our modelling:

- To the very best of our knowledge, none of the related work has performed the task of nowcasting a user’s voting intention over time, under a real-world setting.
- None of the related work has studied the importance of temporal modelling of text – and potentially of other information sources – for this task.
- While the results in several political leaning classification works seem promising, they are often classifying users who are declaring their ideology or stance in their profile, thus the results might be over-optimistic; furthermore, the performance of such models under a realistic evaluation

setting has not been tested yet.

- Almost all past work have employed off-the-shelf models for micro-level tasks related to voting intention or ideology prediction; a better designed model compared to those employing simple feature aggregates can help in boosting the performance for such tasks.

In our modelling, which will be presented in Chapter 6, we account for all of these limitations of previous work in the domain, aiming to build a robust and effective approach that leverages heterogeneous, temporally-sensitive and asynchronous information about the user, in order to nowcast his/her voting intention over time.

3.4 Mental Health and Digital Media (RQ3)

Monitoring mental health with digital media is a field of continuously increasing research interest. Much of this work has focused on using devices such as smart phones, in terms of: *(a)* taking measurements and other data from smart devices, aiming to find correlations between these and some aggregate measurement of well-being, *(b)* classifying social media users or documents with respect to some mental health condition and *(c)* producing models of prediction of mental well-being on the basis of heterogeneous smart phone data in a longitudinal manner. In this section we present examples of leading research in all three categories, whereas in Chapters 7–9 we focus on the latter, which presents a new subfield in the area of mental health monitoring.

3.4.1 Correlation Tasks

Correlation tasks aim to extract features derived from smart devices [204, 170, 177, 256, 6, 149] or social media data [222, 167, 237, 137, 38, 135] from a certain individual, aiming to uncover factors that might be causally linked to their well-being state. Wang et al. [256] revealed a variety of correlations between ac-

tivities derived from smartphone devices of 48 students and their mental state – these include, for example, a significant negative correlation between sleep duration or conversation frequency during the day and depression; Osmani et al. [177] investigated correlations between physical activity and depression in a cohort of bipolar users; Moturu et al. [170] studied correlations between sociability (as derived from Bluetooth proximity data), self-reported sleep times and mood, showing strong relationships between sleep as well as overall sociability and mood; Mehrotra et al. [149] found moderate correlations between various notification- and phone usage-related features, such as the number of applications used or clicks on the screen, with the depressive states of 25 subjects; an analysis of the behaviour of phone usage of bipolar users was performed by Alvarez-Lozano et al. [6] showing strong correlations between app usage and their mood.

Social media correlation studies have mostly focused on the association between the presence of words in dictionaries, such as the LIWC [187], and scores in psychological scales [222, 237, 137]. Others have also included the time frame (e.g., frequency of social media updates) in their analysis [167, 137, 38], showing for example that the frequency of social media interactions is positively correlated to psychological distress [38]. Another perspective of the problem was studied by Lin et al. [135], who worked on social media network structure and showed that both the density and the size of the users’ personal networks were associated with their emotional disclosure.

3.4.2 Stand Alone User or Text Classification

Most works have focused on classifying a piece of text with respect to a certain mental health condition (post-level), or on predicting a mental health condition of some individuals (user-level) based on their social media data (e.g., Twitter, Reddit², ReachOut³, etc.). Typically, a set of social media users with a mental

²<https://www.reddit.com/>

³<https://au.reachout.com/>

condition is identified and matched against a control group. Every individual serves as an instance for a classification task, whose features are extracted from his/her social media timelines. Examples of such tasks involve separating users with post-traumatic stress disorder (PTSD) or obsessive-compulsive disorder (OCD) against a control group, or using social media posts to predict a mental health-related indices of a user (e.g., Satisfaction with Life [129] or PERMA [75] scales). Table 3.1 provides an overview of past works in this category. While most of these works employ longitudinal textual social media data to classify a user’s mental state, they lack of the longitudinal nature of the target (e.g., Depression Level through time) is opposing our goal of automatically assessing mental health in a such a manner.

3.4.3 Longitudinal Models for Assessing Mental Health

During the latest years, more emphasis is put on the importance of assessing mental health in a longitudinal basis. While longitudinal correlation studies can play a vital role in understanding what types of features might be useful for real-time mental health monitoring, it is the *prediction tasks* that have the potential to achieve the goal of mental health monitoring. Such research aims to make use of relevant features extracted from various modalities, in order to train models for automatically predicting a user’s mental state (target), either in a classification or a regression manner [25, 23, 24, 36, 134, 242, 104, 103]. Examples of state-of-the-art works in this domain are listed in Table 3.2, along with the number of subjects that was used and the method upon which evaluation took place:

- *LOUOCV* refers to the leave-one-user-out cross-validation approach (i.e., training on the instances derived from $N - 1$ users and apply the trained model on the left-out user);
- *LOIOCV* refers to the within-user, leave-one-instance-out cross-validation method (i.e., training N different models – one per user – and evaluating

Table 3.1: Works on classifying social media posts/users with respect to mental health conditions.

Work	Description	Data, Analysis
Harman et al. [91]	Classifying PTSD users against a Control group	Twitter User-level
De Choudhury et al. [57]	Develop a Depression Index based on Social Media posts	Twitter Post-level
De Choudhury et al. [58]	Predict future depressive state of an individual (Depression vs Control)	Twitter User-level
Schwartz et al. [219]	Predicting the depression level (as defined by replies to depression-related questions) of a social media user	Facebook User-level
Coppersmith et al. [50]; Resnik et al. [209]; Preotiuc et al. [201]; Pedersen [184]	CLPsych 2015 shared task: binary classification tasks (PTSD, Depression, Control)	Twitter User-level
Coppersmith et al. [49]	Binary classification tasks for 10 mental health conditions (e.g., OCD, PTSD, Bipolar) against a Control group	Twitter User-level
Mitchell et al. [157]	Classification task, identifying users with schizophrenia (Schizophrenic vs Control)	Twitter User-level
Preotiuc et al. [198]	Study the role of inferred personality and demographics for binary mental health classification tasks (PTSD, Depression, Control)	Twitter User-level
Balani & De Choudhury[15]	Predict level of disclosure (No, Low, High) in online social media posts related to mental health	Reddit Post-level
Milne et al. [155]; Brew [33]; Kim et al. [146]; Malmasi et al. [147]; Cohan et al. [43]	CLPsych 2016 shared task: classify mental health posts wrt their severity (Crisis, Red, Amber, Green)	ReachOut Post-level
Coppersmith et al. [51]	Analyse the language of users who committed suicide (Suicidal vs Control)	Twitter User-level
Schwartz et al. [220]	Predicting Satisfaction with Life (SWL) and PERMA indices using online social media at user-/post-level	Facebook Post-level User-level
De Choudhury et al. [59]	Identify users who will transit from discussing about mental health issues online to discussing about suicidal ideation	Reddit User-level
Bagroy et al. [14]	Develop a mental health index for college campus	Reddit Post-level
Benton et al. [18]	Multi-task learning for various mental health conditions (e.g., Neurotypicality, Anxiety, Depression)	Twitter User-level
Amir et al. [7]	Construct latent user representations for classifying a user wrt his/her mental condition (PTSD, Depression, Control)	Twitter User-level

each of them on a leave-one-instance-out manner);

- *MIXED* refers to the randomised validation, by mixing all instances from all users together and forming a randomised K-fold cross-validation setup.

LiKamWa et al. [134] trained models to assess mood – defined by self-reported activeness and pleasure scores in the range [0-4] – based on different smartphone-derived features (e.g., number of emails, visited locations, etc.) in both *LOIOCV* and *LOUOCV* settings; Canzian and Musolesi [36] extracted mobility features based on GPS and accelerometer to train personalised models (*LOIOCV*) that predict depression rates and early signs of depression –as

revealed through daily self-reported PHQ-8 replies– of 28 individuals. Using the *MIXED* evaluation setting, the works by Jaques et al. in [104] and [103] exploited physiology, mobility, survey and phone-related features to predict students’ happiness, alertness, stress, energy and health levels; similarly, Bogomolov et al. extracted proximity (based on Bluetooth) and phone (calls and SMS) features and used them alongside meteorological data and personality traits for predicting happiness [25] and self-reported stress levels ([1-7] scale) of 117 individuals [24], in a randomised 10-fold cross-validation evaluation setting.

To the best of our knowledge, the textual modality has not been explored alongside the various smart phone derived features for this task. Incorporating such heterogeneous and asynchronous information derived from the users’ social media and SMS messages can help in building more robust and accurate models for the task of assessing mental health on a longitudinal manner, as we will showcase in Chapter 7.

Table 3.2: Works on predicting mental health in a longitudinal manner.

Work	Target	Modalities	Type	Size	Eval
Ma et al. [144]	Displeasure Tiredness Tensity (1-5)	location, accelerometer, sms, calls	Class.	15	N/A
Bogomolov et al. [25]	Happiness (1-7)	weather, calls, sms, bluetooth, Big Five	Class.	117	MIXED
LiKamWa et al. [134]	Pleasure (1-5) Activeness (1-5)	email/sms/phone contacts, websites, locations, apps	Regr.	32	LOIOCV LOUOCV
Bogomolov et al. [24]	Stress (1-7)	weather, calls, sms, bluetooth, Big Five	Class.	117	MIXED
Canzian & Musolesi [36]	PHQ-8	GPS	Class.	48	LOIOCV
Jaques et al. [103]	Happiness (0-100)	electrodermal activity, calls, accelerometers, sms, surveys, phone usage, locations	Class.	68	MIXED
Jaques et al. [104]	Happiness Health Alertness Energy Stress (0-100)	social media posts/messages, sms, locations, wifis, charger, headphones, headset, calls, screen/ringer mode, bluetooth	Class.	68	MIXED
Farhan et al. [73]	PHQ-9	GPS, PHQ-9 scores	Class.	79	
Wang et al. [255]	Positive (0-15) Negative (0-15) Positive-Negative	GPS, accelerometer, calls, microphone, light sensor, sms, phone locked, apps	Regr.	21	LOIOCV LOUOCV
Servia-Rodriguez et al. [221]	Positive/Negative Alert/Sleepy	microphone, accelerometer, calls, sms	Class.	726	MIXED
Suhara et al. [229]	Mood (binary)	daily surveys	Class.	2,382	LOUOCV (5-fold CV)

Also, as shown in Table 3.2, most approaches have used the randomised,

user-agnostic (“*MIXED*”) approach to evaluate their models. While tempting, we will show in Chapter 8 that such approaches are vulnerable to bias, thus not necessarily guaranteeing generalisation of their findings. *LOIOCV* approaches that have not ensured that their train/test sets are independent are also vulnerable to bias, if their goal is to create generaliseable personalised models for mental health monitoring in a realistic setting. From the longitudinal works listed in Table 3.2, Suhara et al. [229] performed evaluation in a large dataset using a leave-N-users-out cross-validation approach, thus achieving unbiased results with respect to model generalisability; however, the features employed for their prediction task are derived from self-reported questionnaires of the subjects and not automatically derived.

Building models that can generalise to new users or in a personalised manner is of high importance, for two reasons. First, they can be employed in a real-world setting, thus assisting practitioners in their interventions and users in their self-monitoring of their mental health over time. Second, they can offer insights on the types of behaviour that affect our mental health in an unbiased setting, the establishment of which is crucial in order to avoid drawing false conclusions owed to flaws in the experimental setup [61].

Within the micro-level task of nowcasting mental health indices using heterogeneous data, in this Thesis, we present the following:

- in Chapter 7 we provide the first work on leveraging heterogeneous data from social media posts/messages and smart phones of a cohort of users in order to assess their current mental health state;
- in Chapter 8 we alter our evaluation followed in Chapter 7 to mimic a real-world setting and demonstrate that current state-of-the-art approaches in the domain fail to deliver the reported performance, owed to various flaws in the experimental setup.
- finally, in Chapter 9 we provide our preliminary steps towards generating context-based features from the smart devices of different users, as a first

towards tackling the issues presented in Chapter 8.

3.5 Ethical Considerations

As the research opportunities around mining user-generated content grow, so do the ethical concerns attached to them and the pressure towards the need of higher awareness over such issues [233]. The proliferation of data and the research has also triggered the introduction of “data ethics”, defined as “a new branch of ethics that studies and evaluates moral problems related to data, algorithms and corresponding practices in order to formulate and support morally good solutions” [74].

A key issue with respect to such ethical aspects is related to the privacy of the users whose data is processed in our modelling. Respecting the privacy and keeping the anonymisation of our users is particularly important, since there is an increased risk of harm if those are breached [148]. In the work presented in this Thesis, we have ensured to keep up with the privacy issues that naturally arise when dealing with user-generated content, in an attempt to (a) anonymise and (b) protect the users’ privacy and data. In particular:

- All of our data are stored and processed in an anonymised fashion, in secure servers. For ensuring data anonymity in our experiments, we do not make use of any user-specific information. All of the texts are converted to feature vectors (see previous chapter) and the users are assigned a unique ID. Other information, such as real-time location information, that can be a threat to our users’ privacy, are processed in a location-agnostic manner (i.e., the actual {longitude, latitude} values are converted to a location identifier, e.g., “location 1”). The only case where we mine user-specific information besides these limitations is presented in Chapter 6. There, we use specific keywords to identify social media accounts that might be affiliated with a political party. However, (a) we only use the publicly available description and username, as provided by the users themselves,

and (b) we ensure that at the end of this process the identified users of our queries are appropriately anonymised.

- Despite the fact that the datasets presented in this Thesis form valuable resources for research around mining micro-level indicators, we refrain from publishing any data that could reveal the user’s identity or sensitive user-specific indices, such as a user’s political preference. For example, we refrained from publishing the content of the tweets and the labels of the users that have been employed for our micro-level stance detection task, since the content of the tweets could easily reveal the identity of the user, even if he/she is anonymised – one could still look up for the text on Twitter platform and therefore extract the user. Furthermore, we ensure that any insights that we provide in all of our chapters (e.g., visualisations of user online behaviour) are presented in such a way so that the identity of the user cannot be revealed.
- For our micro-level task of assessing mental health in a longitudinal fashion, we got IRB approval for the study, while ethical consent was provided by all of the participants that took part in it. For the macro- and micro-level political monitoring using social media, we ensure that we comply with Twitter API guidelines. In these cases we only make use of publicly available data that are returned to us through Twitter Streaming API.

Despite addressing those issues, primarily related to the privacy of the users whose content is analysed in this Thesis, there are still several ethical implications that need to be addressed in order for such algorithms to be employed in the real-world. While this is not an issue that is encountered in this Thesis, further ethical issues related to need to also be accounted for such a purpose. For example, several concerns are raised about potential effects of such algorithms, including increased social discrimination and surveillance, decrease of privacy and the introductions of new ways of controlling the public [116]. Such issues of major concern need to be tackled appropriately and proposed directions to-

wards greater transparency, accessibility, public supervision and regulation of data mining models and practices need to be taken into account when employing such models in the real-world [116].

Part II

Macro-level Modelling Using Social Media

CHAPTER 4

Building and Evaluating Sentiment Analysis Resources

Preliminary analysis. *Building and evaluating natural language resources for the tasks of sentiment analysis, emotion (affect) analysis and sarcasm detection over social media content.*

The current chapter focuses on our document-level preliminary analysis and presents the first steps towards monitoring opinion, as expressed in online social media. In particular, here we focus on the static tasks on *sentiment analysis*, *emotion (affect) analysis* and *sarcasm detection*. We work in an under-resourced language (i.e., the Greek language) to generate NLP resources from scratch and test their effectiveness and robustness with a primary focus on the cross-domain sentiment analysis task. The experiments using different algorithms and parameters on our resources show promising results over standard baselines; on average, we achieve a 24.9% relative improvement in F-score on the cross-domain sentiment analysis task when training the same algorithms with our resources, compared to training them on more traditional feature sources, such as n-grams. Importantly, the generated resources also show promising results in related tasks, such as emotion analysis and sarcasm detection. This kind of evaluation over multiple tasks is essential in order to build resources that can be used in various macro- or micro-level tasks over user-generated content¹.

¹The current chapter is based on [245].

4.1 Introduction

During the last decade, the amount of content that is published online has increased tremendously, primarily due to the wide adoption and use of Online Social Media (OSM) platforms. The content produced within OSM has the potential to be used for understanding, modeling and predicting human behavior and its effects. Unsurprisingly, OSM mining has been used in this sense for various tasks, such as trend detection [2], crime rates [82] and election results prediction [243], tracking influenza rates [124] and others.

A key task that often needs to be dealt within such problems is *sentiment analysis* – the task of classifying a piece of text with respect to its sentiment, which can be positive, negative or neutral. Other closely related tasks also include *emotion (affect) analysis* and *sarcasm detection* [88]. All these tasks are fundamental in order to understand and analyse the public sentiment, emotion or stance around current events and topics of public debate. Despite the fact that a lot of research works on sentiment analysis rely primarily on sentiment lexicons [64, 230, 171, 160, 259], there is not (to the best of our knowledge) any *large-scale* and *systematically evaluated* lexicon for the Greek language. The case of the Greek language, as expressed in social media, is particularly interesting for being studied under these tasks, owed to several challenges that arise: works in other languages that create sentiment resources based on SentiWordNet [71] and WordNet synsets [154] are not applicable to noisy, user-generated content, such as that of OSM; other works making use of syntactic or Part-of-Speech (POS) resources [250, 106] cannot be applied on the Greek language, due to the insufficient accuracy of the relevant tools (POS taggers) for Greek. Furthermore, most of the past works evaluate their created resources in a manual fashion, or in a single task (e.g., sentiment analysis); however, real-world multi-task and multi-domain evaluation of sentiment-related resources and comparison with well-established feature baselines are needed in order to demonstrate their effectiveness and generalisation capabilities, as well as their

potential weaknesses.

In the current chapter, we overcome the difficulties stemming from the limited availability of linguistic resources for the Greek language by building upon the definitions of the Greek lemmas of a general lexicon; we present the first publicly available manually annotated Greek Affect and Sentiment lexicon (“GrAFS”); we adapt past methodologies for the English language [160, 259, 203] and, based on our annotations, we create two separate large-scale lexicons for sentiment analysis on social media. We expand our resources based on recent developments in the field of Natural Language Processing, by creating *word embeddings* representations [85]. We move well beyond the manual evaluation of our resources and provide in-depth analysis of their effectiveness in three different tasks (sentiment and emotion analysis [161], sarcasm detection) in various datasets using different approaches. Finally, we make all of our resources publicly available for the research community².

4.2 Generating the Resources

Here we present the three lexicons that have been created. We first present the manually annotated lexicon (“GrAFS”) that was generated using the online version of Triantafyllides’ Lexicon [239], as a starting point (section 4.2.1). Then, we present the automatically generated sentiment lexicons (4.2.2) and the word embeddings representations (4.2.3).

4.2.1 GrAFS Lexicon Creation

The lexicon by Triantafyllides [239] is one of the largest and widely recognised general dictionaries existing for the Modern Greek language, counting 46,747 lemmas. One of its distinctive features is that, despite the fact that it has been designed for human use, it seems to have been conceived to promote NLP tasks, as it standardises linguistic data (e.g., nouns are organised in declension classes,

²The resources are available at: mklab.iti.gr/resources/tsakalidis2017building.zip

descriptions are given in a systematic way, without comments or assumptions). Furthermore, in its electronic version, as provided by the Centre for the Greek Language³, all information types are tagged (e.g., part of speech, declension class, example, etymology, use, register of language, semantic field), making it the largest existing lexical resource of that type for use in NLP tasks in the Greek language. In order to aggregate words that could possibly contain sentimental load, we crawled the electronic version of the lexicon. In particular, we used the advanced search utilities to retrieve all words that can be used in an ironic (346 words), derogatory (458), abusive (90), mocking (31) or vulgar tone (53). Furthermore, since the electronic version of this lexicon provides the capability to search through the description of every word, we further searched these descriptions for emotional words (e.g. *feel*)⁴.

The above process resulted in the collection of 2,324 words and their definitions. Those were then manually annotated with respect to their expressed sentiment and affect by four annotators – two with a Computer Science and two with a Linguistics background. Every annotator was first asked to annotate each word as **objective**, or **strongly** or **weakly subjective**. If subjective, then the annotator would assign a polarity label to the word (**positive/negative/both**) and rate it with respect to its affect in an integer scale from 1 (does not contain this affect at all) to 5 along Ekman’s six basic emotions (**anger, disgust, fear, happiness, sadness, surprise**) [68]. In all annotations (subjectivity, polarity and the six emotions), the annotators were allowed not to rate a word at all if they were not sure about its meaning and use. We also created extra columns for comments and proposed synonyms for every word, but did not use those fields for the purpose of this work. These annotations have been previously released; however, no systematic evaluation has been performed on them up to now. The complete instructions that were provided to the annotators can be found in Appendix A (translated to English).

³http://www.greek-language.gr/greekLang/modern_greek/tools/lexica/index.html

⁴The exact words that were used and the number of words found are: *συναίσθημα* (603), *αισθάνομαι* (154), *αίσθηση* (121), *αίσθημα* (793), *συναίσθηση* (17), *αισθάνεται* (88), *νιώθω* (59).

Table 4.1: Annotators’ agreement for subjectivity (Pearson Correlation), positive and negative (Cohen’s Kappa), respectively.

subjectivity				positive				negative			
	#2	#3	#4		#2	#3	#4		#2	#3	#4
#1	.47	.90	.77	#1	.40	.82	.51	#1	.28	.85	.45
#2		.45	.59	#2		.38	.45	#2		.31	.42
#3			.60	#3			.53	#3			.47

Table 4.2: Annotators’ agreement (Pearson Correlation) for the six emotions.

anger				disgust				fear			
	#2	#3	#4		#2	#3	#4		#2	#3	#4
#1	.28	.68	.55	#1	.47	.74	.57	#1	.37	.60	.35
#2		.34	.39	#2		.45	.53	#2		.41	.28
#3			.58	#3			.56	#3			.46

happy				sad				surprise			
	#2	#3	#4		#2	#3	#4		#2	#3	#4
#1	.42	.83	.62	#1	.40	.59	.47	#1	.18	.50	.17
#2		.40	.53	#2		.39	.46	#2		.18	.40
#3			.62	#3			.53	#3			.20

Then, we eliminated words for which there was a missing subjectivity score for more than one annotator, reducing our lexicon to 2,260 words. We corrected the few entries that were judged as **objective** but had a non-zero polarity or emotional score, by converting the positive and negative scores to 0 and the emotion scores to 1 (that is, their minimum allowed score), since these entries were judged to be wrongly annotated, as they were not in line with the annotation instructions. We also converted the subjectivity scores to three values: 0 for objective, .5 for weakly subjective and 1 for strongly subjective. Finally, we averaged the subjective, positive, negative and the six emotion scores as provided by the annotators. The annotators’ agreement is shown in Tables 4.1 and 4.2. We measure the agreement in terms of Cohen’s Kappa for the positive and negative dimensions, since these form two distinct classes; for the rest, we measure the agreement in terms of Pearson correlation. We notice a fair agreement (.40-.60) in most cases, with the exception of the surprise dimension. The reason behind this is probably the nature of the surprise emotion, which, in contrast to the rest, can be expressed both in a positive and negative way, thus challenging the annotators.

Since the Greek language is a highly inflected language, the next step was to produce all inflected forms derived from the extracted lemmas. This task was performed semi-automatically, using NLP tools developed by the Laboratory of Translation and Natural Language Processing for Greek language analysis [48, 123], thus expanding the list of our keywords using all declension and conjugation classes derived from the original words and replicating their sentiment and emotion scores. The final version of the lexicon after this process consists of 32,884 unique inflected forms⁵. Figure 4.1 displays the distributions of the scores before and after the morphological expansion (for the six emotions, we normalised the scores in the [0,1] range). What is noticeable is that the distributions are not affected by the expansion: the lower Pearson correlation between them is observed for the case of “Negative” sentiment (.89); for the rest of sentiments and emotions, the respective correlation is $>.95$. Furthermore, it is shown that there are more negative than positive words, while the majority of the words do not carry a strong emotional value, as indicated by the annotators.

4.2.2 Twitter-Specific Sentiment Lexicons

A common drawback of applying a sentiment lexicon in user-generated content is that, due to the informal nature of the content, it is difficult to find exact matches of the keywords in the lexicon. For that reason, we created two Twitter-specific lexicons that have the potential to capture a larger portion of sentiment-related keywords as expressed in social media, including misspellings, abbreviations and slang.

Given a set of positive (D_{pos}) and negative (D_{neg}) documents composing a corpus D with $D_{pos} \cup D_{neg} = D$ and $D_{pos} \cap D_{neg} = \emptyset$, as discussed in Chapter 2, a common practice to find the degree of association of each n-gram n appearing in D with each sentiment class (pos, neg) is to calculate the Pointwise Mutual Information (PMI) of n with respect to each class and use Eq. 4.1 to assign a

⁵In cases of duplicated words owed to the expansion, we only kept their first occurrence.

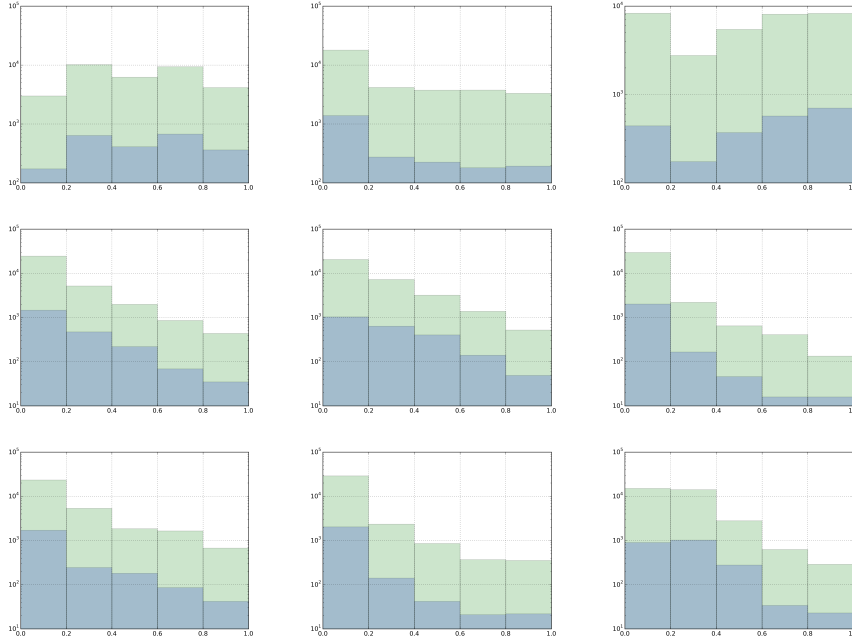


Figure 4.1: Distributions (in log scale) of word scores before (blue) and after (green) the morphological expansion (from top-left to bottom-right: subjective, positive, negative, angry, disgust, fear, happy, sad, surprise).

score sen to it [160]:

$$sen(n) = PMI(n, pos) - PMI(n, neg), \quad (4.1)$$

where $PMI(n, cls) = \log(p(cls|n)/p(cls))$ for each class $cls = \{pos, neg\}$. This process results in a dictionary that associates each n-gram with a sentiment score. Then, feature extraction from a document can take place based, for example, on the summation of the n-grams' sentiment scores. While the lexicons that have been created for the English language using this methodology have proven to be quite effective [160, 259], the task of creating a large-scale annotated Greek corpus to serve as D is quite difficult and time consuming. To deal with this issue, we used two semi-supervised methods and created two Twitter-specific lexicons. For both, we used the Twitter Streaming API⁶, in order to collect tweets in the Greek language. Then, we followed some common prepro-

⁶<https://dev.twitter.com/streaming/overview>

cessing steps (tokenisation [83], lowercasing, replacement of user mentions with `usrmention` and of URLs with `urlink`, removal of non-alphanumeric characters and of one-character-long unigrams) and calculated the score of every n -gram appearing at least 10 times in D , according to Eq. 4.1.

Keyword-Based Lexicon (KBL) We collected about 15 million tweets in Greek (excluding retweets) over a period of more than two months (August–November 2015) constrained on the occurrence of at least one of 283 common Greek stop words⁷. Constraining our search on such a list (instead of constraining strictly on the Greek language) was a step towards (a) filtering out false positive tweets returned by Twitter API (e.g., tweets containing a few Greek letters that might be wrongly identified as “tweets written in Greek”) and (b) aggregating tweets that have potentially some meaningful textual content. In order to create our corpus D , positive and negative words from GrAFS were used as seeds. This stems from our assumption that a tweet containing a polarised keyword would lead to the respective sentiment for the whole tweet. We consider a positive (negative) word as a positive (negative) seed word if (a) its subjectivity score in the GrAFS lexicon is at least 0.75, (b) its positive (negative) score is 1.0 and (c) its negative (positive) score is 0. In this way, we extracted words with clearly positive and negative sentiment (based on our annotations), ending up with 1,807 positive and 4,852 negative seed words. Intuitively, relaxing the previous constraints would yield more, yet noisier, seed words; for that reason, we avoided using such an approach. Using our seed words, and not taking into consideration the short tweets in our collected data ($length < 25$ characters), we found 593,321 positive and 340,943 negative tweets in our corpus. We excluded tweets appearing in both positive and negative tweet sets, resulting in a dataset of 892,940 tweets to be used as the corpus for generating our first Twitter-based lexicon. After the preprocessing steps mentioned above, we were left with 190,667 n -grams (52,577 unigrams, 138,090 bigrams) comprising our

⁷The Streaming API receives a list of keywords and a language specification as input.

Keyword-based lexicon (KBL).

Emoticon-Based Lexicon (EBL) A practice that is commonly followed in sentiment analysis in OSM in order to create large-scale training sets is to search for tweets containing emoticons and assign them the corresponding sentiment or emotional label [84, 203, 244]. We followed this procedure, collecting tweets containing emoticons of the six basic emotions [68] as in [203], over a period of five months (January–June 2015). Only tweets containing happy- and sad-related emoticons were in reasonable quantity to serve our purposes (about 200K/25K tweets with happy/sad emoticons, respectively), under the restrictions of being non-retweeted tweets and of a minimum length of 25 characters. Following the exact same procedure as with the KBL lexicon, we created the new lexicon (EBL) containing 32,980 n-grams (14,424 unigrams, 18,556 bigrams).

The method for creating the two Twitter-based lexicons is the same (only the corpus changes). Indeed, we found that 88% of the n-grams that are included in EBL, are also present in KBL. Interestingly, the Pearson correlation between the co-occurring terms is only 29.5%. The reason for this is that the corpus of creating the EBL lexicon is noisier and smaller compared to the KBL. In an attempt to quantify the noise contained in our lexicons, we compiled a list of 634 stop words⁸ and found that many of them are included in our lexicons with some sentiment score (485 in KBL; 414 in EBL). Other cases, such as negation, are also not explicitly handled by our lexicons. For example, 1.9% of the entries in KBL (2.7% in EBL) are n-grams that contain one of the five most popular negation words in Greek ($\mu\eta(\nu)$, $\delta\epsilon(\nu)$, $\acute{o}\chi\iota$), with the majority of them (62% in KBL; 70% in EBL) having negative scores. We consider dealing with such linguistic cases as part of our future work.

⁸Available through <http://www.translatum.gr>.

4.2.3 Twitter-Specific Word Embeddings

While sentiment lexicons have shown a great potential when applied on OSM data, they still do not capture the context of a keyword: a sentiment score is assigned to every n-gram, regardless of the context it is being used. Most importantly, n-grams are represented as different discrete symbols, providing us with no information of the similarity of their meaning. To address this limitation, dense word representations have been proposed to capture the context in which they appear and have gained ground over the latest years [248]. Recent advances have made it possible to tackle this problem by representing every word as a vector of values (“word embedding”), which is generated through various methods, such as neural networks or dimensionality reduction on the word co-occurrence matrix [153, 152, 85].

To assess the effectiveness of such representations in the Greek language, we applied word2vec using the skip-gram architecture [153] in our corpus of 15M tweets that was used for creating KBL⁹. The selection of word2vec was based on its wide and successful application in many NLP tasks, while the selection of the skip-gram architecture was based on its ability to deal with rare dictionary words that appear quite often in social media due to their noisy nature. We followed the same pre-processing steps as with our lexicons, set the minimum frequency of unigrams to 5 and used a 5-token window around every word. We opted for a smaller number of word occurrences compared to the lexicons (5 vs 10) since word2vec produced context-aware word representations, thus requiring smaller number of training examples compared to the co-occurrence-based method of generating our lexicons. Then, we created word embeddings of length $n = 300$ ($|V| = 418,402$). Further increasing the length of the vector representations would have led to a high increase in computational cost during the learning process, while there is not sufficient evidence in literature that a larger length would also imply an increase in accuracy for sentiment-related tasks.

An alternative way of generating such latent representations would have been

⁹The Python package gensim was employed (<https://pypi.python.org/pypi/gensim>).

to train a neural network on a labeled (positive/negative) corpus [114] – e.g., by using the corpus used for EBL with positive/negative emoticons. However, this would have been based on a much smaller corpus, resulting in task-specific representations that might not be as effective in other tasks. We have also tried to build representations derived from word2vec using the sentiment-specific corpora from which our lexicons were built; however, we noticed that the accuracy dropped in the experiments that follow in the next sections, compared to the one obtained by using the full-corpus word2vec representations. The reason for this is that the sizes of the corpora that were used for creating the KBL/EBL lexicons were much smaller than the 15M tweets corpus (890K/225K, respectively), thus providing word2vec with much less contextual information about the words, leading into qualitatively poorer word embeddings representations.

4.3 Experimental Setup

To evaluate our resources, we performed several experiments, using different algorithms on three different sentiment-related tasks, as follows:

- **Task 1 (Sentiment Analysis):** Given a tweet, classify it as positive, negative or neutral (classification task).
- **Task 2 (Emotion (Intensity) Analysis [161]):** Given a tweet, find the level for each of the conveyed emotions, on a 0-5 scale (regression task).
- **Task 3 (Sarcasm Detection):** Given a tweet, classify it as being sarcastic or not (binary classification task).

4.3.1 Datasets

Task 1 We worked on three different datasets for the sentiment analysis task, as presented in Table 4.3. The first two (“TIFF”, “TDF”) were acquired from Schinas et al. [218] and consist of tweets in Greek and English, concerning the Thessaloniki Film Festival and Thessaloniki Documentary Festival respectively.

Table 4.3: Number of tweets per-class in the sentiment analysis task.

	positive	neutral	negative	total
TIFF	876	1566	314	2756
TDF	786	813	228	1827
GRGE	79	979	582	1640

In our experiments, we focused strictly on the tweets written in Greek¹⁰. The third dataset (“GRGE”) consists of tweets related to the January 2015 General Elections in Greece, extracted by providing the Streaming API with a keyword list of the main political party names, their abbreviations and some common misspellings. All duplicates were excluded and 2,309 tweets (randomly selected) were annotated with respect to their sentiment. Each tweet was annotated by two MSc graduates (one with Engineering and one with Economics background) and native Greek speakers, who were selected based on their keen interest in the elections in order to ensure good annotation quality. The annotators were asked to detect the sentiment of the author of the tweet. In rare cases of presence of both positive and negative sentiment within the same tweet, the annotators were instructed to annotate it based on the prevailing sentiment. The Cohen’s kappa coefficient over the initial set of 2,309 tweets was 0.525. Hence, we only kept the ones (1,640) for which there was an agreement.

Task 2 For the emotion analysis task we used the dataset made available by Kalamatianos et al. [112]. It consists of 681 tweets annotated by two annotators with respect to their emotion on a scale from 0 to 5. Due to the low agreement between the annotators for the **angry** and **disgust** emotions, we excluded them from our analysis; for the rest, we consider the average emotion score given by the two annotators as our ground truth.

Task 3 To the best of our knowledge, there does not exist a publicly available dataset for sarcasm detection in the Greek language. Therefore, we created a new annotated dataset, consisting of tweets related to the Greek General

¹⁰Language recognition was performed using <https://github.com/saffsd/langid.py>

Elections of January, 2015. A random set of 3,000 tweets were annotated with respect to being sarcastic or not. Every tweet was annotated by the same annotators as the GRGE dataset (**sarcastic/non-sarcastic** – or **N/A**, if the annotator was uncertain); we then removed all the tweets that were marked as **N/A** and only kept the ones for which there was an agreement (2,506 overall, Cohen’s kappa coefficient: 0.76). Note that, as expected, the majority of tweets (79.3%) belong to the **non-sarcastic** class (1,988 vs 518).

4.3.2 Feature Extraction

We used three different sets of features which are extensively used in sentiment-related tasks in the English language. Before performing feature extraction, we applied the same pre-processing steps as for the lexicon generation (lowercasing, replacing URLs and usernames, tokenising and removing all non-alphanumeric characters). Note that some of these steps might actually hurt accuracy in sentiment-related tasks (e.g., an all-uppercase word in a tweet might be indicative of the tweet sentiment); we leave the assessment of such features as part of our future research. We did not perform stop word removal or stemming, since those steps were found to have no or negative influence on the sentiment analysis tasks [213, 20] and we had to be consistent with the way that our lexicons were previously created. The feature sets that were extracted are the following:

- **Ngrams (N)**: For each of our tasks, we extracted unigrams and bigrams with binary values, excluding n-grams that appeared only once in the training set.
- **Lexicons (L)**: We mapped every unigram and bigram to both KBL and EBL and extracted the following features: the number of positive (negative) matches of every unigram and bigram in the lexicons (that is, the total count of unigrams/bigrams with associated lexicon score larger – for positive – and smaller – for negative – than zero), the total sum (float)

of positive (negative) unigrams and bigrams scores and the overall summation of their respective scores. We also extracted the same features regardless of whether they referred to unigrams or bigrams. This led to a total number of 30 features per tweet. Finally, using the initial GrAFS lexicon, we extracted the overall sum of the unigrams’ subjective, positive and negative scores, as well as the six emotions, leading to a total number of 39 features.

- **Word Embeddings (E):** We mapped every word of every tweet to its word embeddings vector. In order to represent every tweet in these vector spaces, we applied three functions on every dimension of its words’ vectors (*min*, *max* and *mean*) [231], leading to 900 features for every tweet. Other functions, such as the summation or the multiplication, could have also been used; however, finding the optimal type of functions to use was considered out of the scope of this work.

Each of these feature sets was examined separately in our experiments. We also created representations, by merging each pair (“NL”, “NE”, “EL”), as well as all of them together (“NLE”). These seven representations were provided separately as input to our classifiers in the three tasks, to examine their effectiveness when used alone and in conjunction with each other. To get further insights on the quality of our resources, we also compare the performance for the same tasks and with the same setup when using features derived strictly from (a) our GrAFS lexicon (“ \mathbf{L}_g ”), (b) the Twitter-specific lexicons (“ \mathbf{L}_{tw} ”) and (c) an automatically translated sentiment lexicon for the English language (“ \mathbf{L}_{tr} ”). For the latter, we employed the popular Emotion Lexicon by Mohammad et al. [163, 164], which contains annotations of English words with respect to 10 affect dimensions (subjective, positive, negative, angry, anticipation, disgust, fear, happy, sad, trust), 7,189 of which have been automatically translated into Greek using Google Translate¹¹. The features are extracted by summing the number of unigram/bigram occurrences for each dimension of every tweet.

¹¹<https://translate.google.com>

4.3.3 Classification and Regression Algorithms

To explore the use of our resources in depth, we employed three algorithms for the classification tasks (Task 1 and 3). These were the Logistic Regression (LR), Random Forests (RF) and Support Vector Machines (SVM) with an RBF kernel. Every algorithm was tested on each set of features for all tasks using 10-fold cross validation. In order to study the cross-domain effectiveness of our features on Task 1, we also performed experiments by training on the feature sets of every two datasets and testing on the third. For the regression task (Task 2), we opted to use the Least Absolute Shrinkage and Selection Operator (LASSO), Random Forests for Regression (RFR) and Support Vector Regression (SVR). Due to the small size of the dataset in Task 2, we opted for a 5-fold cross-validation (to avoid having folds of very small size).

We did not perform parameter optimisation in any of the tasks, as finding the optimal parameters or algorithms was out of the scope of the current work; however, we did run our experiments with different parameters (the α parameter for LASSO, the number of trees for RF/RFR and the C parameter in SVM/SVR). For LASSO, we performed our experiments with different values for the α parameter ranging from 10^{-5} to 10^3 ; for SVM and SVR we performed experiments with C varying from 10^{-5} to 10^3 ; for RF and RFR, we performed our experiments with 100 up to 1,000 trees, with increases of 100. Only the results of the algorithms with the best-performing parameters are reported; however, there were not major deviations in the results of any algorithm under different parameters observed in any task (except for extreme cases of C in SVM/SVR).

Owed to the lack of a well-established state-of-the-art approach in the under-resourced Greek language, we have selected to compare the performance of the algorithms trained on our resources against some naïve baselines, in order to get insights on their effectiveness. In particular, we have compared the results obtained by the classification algorithms (Tasks 1, 3) against the majority class baseline (MC). For the regression task (Task 2), we defined our baselines as

(a) the average ground-truth predictor MC_{avg} and (b) the model MC_{dist} that predicts an emotion score for an instance randomly, yet based on the probability distribution of the ground-truth; for the latter, we performed 1,000 experiments and report here average statistics for every emotion.

4.4 Results

4.4.1 Task 1: Sentiment Analysis

We used the weighted-average F-measure for the evaluation of Task 1. This was selected due to its nature of being a harmonic mean between precision and recall, while weighted-averaging was preferred over macro-averaging, in order to avoid a biased estimation of the algorithms' performance, due to the limited amount of positive examples in the GRGE dataset. Results are presented per dataset and per algorithm, as well as macro-averaged (across the three datasets). We are also presenting the Majority Classifier (MC) as our baseline.

Table 4.4 presents the results obtained using 10-fold cross validation on the three datasets. The comparison between our two lexicons shows that our expanded L_{tw} lexicon captures domain-specific sentiment features better than L_g , probably due to its larger size, whereas better performance is achieved consistently on average when these two resources are merged (L). Importantly, all of our lexicon resources outperform the translated L_{tr} lexicon by a clear margin. From the six individual representations, n-grams (N) and word embeddings (E) consistently outperform all the lexicon-based representations. Despite that, our lexicons can be used effectively alongside with both representations, yielding a slightly better performance than the individual L/E models. However, the main advantage of the lexicon (L) and word embeddings (E) representations is their cross-domain nature, which is studied next.

The domain-dependence of the n-grams representation (N) is clearly illustrated in Table 4.5. For comparison purposes, we have also included the relative decrease obtained in the cross-domain experiments when compared to the corre-

Table 4.4: F-measure based on 10-fold cross-validation for Task 1.

dataset	model	baselines		our resources				combinations			
		N	L_{tr}	L_g	L_{tw}	L	E	NL	NE	LE	NLE
TIFF	MC	41.15	41.15	41.15	41.15	41.15	41.15	41.15	41.15	41.15	41.15
	LR	61.35	42.75	55.32	56.29	57.83	59.56	63.29	60.28	62.28	62.49
	RF	56.93	44.20	57.99	56.08	59.54	59.79	59.90	59.00	61.51	60.62
	SVM	59.52	43.99	58.00	48.31	49.73	61.96	62.11	62.53	63.58	64.34
TDF	MC	27.36	27.36	27.36	27.36	27.36	27.36	27.36	27.36	27.36	27.36
	LR	62.64	42.48	51.22	53.87	54.17	60.56	65.87	62.27	61.86	63.23
	RF	58.85	45.96	52.05	54.67	59.18	62.40	62.45	62.42	63.97	63.85
	SVM	60.24	46.05	51.64	53.65	53.75	63.29	63.75	63.22	65.28	66.53
GRGE	MC	44.63	44.63	44.63	44.63	44.63	44.63	44.63	44.63	44.63	44.63
	LR	80.37	52.11	60.86	72.52	72.46	76.72	80.66	77.82	77.55	78.06
	RF	79.35	53.35	65.32	71.43	73.19	78.14	76.42	78.01	78.28	77.98
	SVM	79.17	52.82	62.76	68.30	68.44	80.65	79.36	79.71	80.32	79.72
avg	MC	37.71	37.71	37.71	37.71	37.71	37.71	37.71	37.71	37.71	37.71
	LR	68.12	45.78	55.80	60.89	61.49	65.61	69.94	66.79	67.23	67.93
	RF	65.04	47.84	58.45	60.73	63.97	66.78	66.26	66.48	67.92	67.48
	SVM	66.31	47.62	54.47	56.75	57.31	68.63	68.41	68.49	69.73	70.20

sponding intra-domain ones that were presented in Table 4.4. The performance of our algorithms when trained on n-grams from the other two datasets drops by 28.29% on average, compared to the 10-fold cross-validation approach. This highlights the importance of using features that can be used in a cross-domain fashion, so that one does not need manually annotated data for all possible domains, in order to develop an accurate sentiment classifier. L_{tr} can barely outperform the majority classifier (MC); on the contrary, our manually annotated L_g lexicon is the most robust representation. Word embeddings form again the best-performing individual feature set, followed by our lexicon-based features. Those two combined (LE) yield the best across-algorithm and across-datasets results; the incorporation of n-grams on top of them has a slightly negative effect on the performance on average (except for the case of SVM). This is an important finding for the cross-domain sentiment analysis task also, because it indicates that the use of a relatively small, fixed number of features can yield better results, alleviating the learning models from the task of dealing with the sparse bag-of-words representations that have a negative effect on the accuracy, while increasing the computational cost. Finally, it should be noted that the accuracy of the best performing feature set in the GRGE dataset drops much more than the accuracy on TDF and TIFF, if we compare those against the

results obtained by 10-fold cross-validation (from 80.66 to 63.71). The reason behind this effect is that the TDF/TIFF datasets are related (documentary and film festivals respectively), as opposed to the GRGE. Thus, the performance achieved in GRGE represents a more realistic evaluation of our resources in a completely new domain.

Table 4.5: F-measure based on cross-domain experiments for Task 1. The first column indicates the test dataset, after training the models on the rest.

test set	model	baselines		our resources				combinations			
		N	L_{tr}	L_g	L_{tw}	L	E	NL	NE	LE	NLE
TIFF	MC	41.15	41.15	41.15	41.15	41.15	41.15	41.15	41.15	41.15	41.15
	LR	53.56	42.58	57.88	57.54	58.43	58.90	59.93	58.26	60.20	58.46
	RF	54.55	44.74	56.68	55.32	57.20	62.64	60.08	61.35	63.73	63.00
	SVM	51.42	44.20	57.14	47.49	49.47	60.45	61.56	61.09	61.30	63.32
TDF	MC	27.36	27.36	27.36	27.36	27.36	27.36	27.36	27.36	27.36	27.36
	LR	44.01	28.81	44.45	50.41	51.96	56.11	59.81	54.14	57.28	56.17
	RF	34.20	31.37	47.40	50.40	53.02	50.86	49.16	43.85	54.76	46.34
	SVM	40.68	31.30	47.38	36.57	38.06	59.03	56.42	59.51	59.51	61.02
GRGE	MC	44.63	44.63	44.63	44.63	44.63	44.63	44.63	44.63	44.63	44.63
	LR	51.14	45.79	49.20	56.63	56.49	60.06	55.90	56.43	61.32	59.22
	RF	46.17	46.62	49.85	58.03	58.97	48.27	52.84	48.46	51.27	48.13
	SVM	53.56	46.38	51.61	45.68	47.31	63.71	62.01	63.19	57.07	63.04
avg	MC	37.71	37.71	37.71	37.71	37.71	37.71	37.71	37.71	37.71	37.71
	LR	49.57	39.06	50.51	54.86	55.63	58.36	58.55	56.28	59.60	57.95
	RF	44.97	40.91	51.31	54.58	56.40	53.92	54.03	51.22	56.59	52.49
	SVM	48.55	40.63	52.04	43.25	44.95	61.06	60.00	61.26	59.29	62.46
relative decrease (%)	LR	27.23	14.68	9.48	9.90	9.53	11.05	16.29	15.74	11.35	14.69
	RF	30.86	14.49	12.22	10.13	11.83	19.26	18.46	22.95	16.68	22.21
	SVM	26.78	14.68	4.46	23.79	21.57	11.03	12.29	10.56	14.97	11.03

4.4.2 Task 2: Emotion Intensity Analysis

We used the mean squared error (MSE) and Pearson’s correlation coefficient (ρ) as the evaluation measures for this task. These are popular for the evaluation of regression tasks, measuring the error by putting more weight on the larger errors (MSE) and the correlation between the predicted and the actual scores, respectively.

Tables 4.6 and 4.7 show the results using 5-fold cross-validation. “Fear” is the emotion for which all models achieve the lowest error rates, albeit barely outperforming our baseline model MC_{avg} ; Pearson correlation is also low, due to the low variance of values in the dataset for this emotion. For the rest of the emotions, the results reveal a similar difficulty level with each other in terms of

predicting their values. In all cases, our features clearly outperform the N and L_{tr} baselines.

For clearer comparison, Table 4.8 presents the cross-emotion results (MSE, ρ); in particular, we present the macro-average evaluation metrics across all algorithms and emotions, as well as the macro-average metrics, by selecting the best algorithms per emotion and representation (e.g., SVR’s $\rho = .388$ is selected against LASSO and RFR for the “happy” emotion for the N representation). Intuitively, the selection of the best algorithm for every emotion is crucial in a real-world application, thus the comparison of the best algorithms per representation in Table 4.8 is of great importance.

The comparison between the different features reveals that the lexicon features L_{tw} and L clearly achieve the lowest error rates on average; however, it is the word embeddings and the combined representations using them that outperform the rest with respect to ρ . Note that the MC_{avg} has an MSE-average of 1.72, which is equal to the MSE-best of L_{tr} , demonstrating the inability of the latter to capture the emotion contained within a tweet. The comparison between our lexicons shows that L_g performs poorly compared to L_{tw} (probably due to the noisy language of social media, which is better captured by L_{tr}), whereas their combination into L does not boost performance for this task. Overall, the comparison of the best models per emotion and per representation reveals that our word embeddings form the best representation for this task and a small boost in accuracy is provided when our lexicon features are used alongside them (LE). This is an important finding, as it shows that our resources can provide a relative improvement of 13.5% in MSE rates (28.4% in ρ) over the most competitive pre-existing baseline (N), despite the fact that they were built with a primary focus on the task of sentiment analysis.

4.4.3 Task 3: Sarcasm Detection

Table 4.9 presents the F-score on a per-class and a macro-average basis. We include the per-class results, in order to study them in more detail, with an

Table 4.6: MSE for the Emotion Prediction task (Task 2), using 5-fold cross validation.

emotion	algorithm	baselines		our resources				combinations			
		N	L_{tr}	L_g	L_{tw}	L	E	NL	NE	LE	NLE
fear	MC_{avg}	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68
	MC_{dist}	1.35	1.35	1.35	1.35	1.35	1.35	1.35	1.35	1.35	1.35
	LASSO	0.88	0.70	0.69	0.67	0.68	0.98	0.85	0.77	0.98	0.78
	RFR	0.73	0.73	0.73	0.67	0.68	0.71	0.66	0.67	0.70	0.67
	SVR	0.69	0.73	0.75	0.69	0.71	0.67	0.73	0.73	0.66	0.71
	average	0.77	0.72	0.72	0.68	0.69	0.79	0.75	0.72	0.78	0.72
happy	MC_{avg}	2.08	2.08	2.08	2.08	2.08	2.08	2.08	2.08	2.08	2.08
	MC_{dist}	4.17	4.17	4.17	4.17	4.17	4.17	4.17	4.17	4.17	4.17
	LASSO	2.42	2.09	1.93	1.92	1.87	2.61	2.48	2.28	2.60	2.26
	RFR	1.94	2.06	1.87	1.72	1.69	1.57	1.68	1.57	1.56	1.57
	SVR	1.87	2.20	2.05	1.65	1.69	1.62	1.93	1.78	1.62	1.72
	average	2.08	2.12	1.95	1.76	1.75	1.93	2.03	1.88	1.93	1.85
sad	MC_{avg}	1.98	1.98	1.98	1.98	1.98	1.98	1.98	1.98	1.98	1.98
	MC_{dist}	3.98	3.98	3.98	3.98	3.98	3.98	3.98	3.98	3.98	3.98
	LASSO	2.35	2.00	1.92	1.89	1.91	2.80	2.28	2.11	2.80	2.07
	RFR	1.82	2.07	1.95	1.77	1.71	1.58	1.68	1.58	1.58	1.58
	SVR	1.85	2.75	2.87	1.81	1.87	1.65	2.09	1.81	1.66	1.80
	average	2.01	2.27	2.25	1.82	1.83	2.01	2.02	1.83	2.01	1.82
surprise	MC	2.12	2.12	2.12	2.12	2.12	2.12	2.12	2.12	2.12	2.12
	MC_{dist}	4.19	4.19	4.19	4.19	4.19	4.19	4.19	4.19	4.19	4.19
	LASSO	2.82	2.13	2.12	1.96	1.99	3.22	2.75	2.3	3.16	2.28
	RFR	1.82	2.18	2.10	1.72	1.67	1.57	1.63	1.56	1.57	1.56
	SVR	1.87	2.36	2.24	1.88	1.95	1.79	2.02	1.87	1.68	1.82
	average	2.17	2.22	2.15	1.85	1.87	2.19	2.13	1.91	2.14	1.89

emphasis on the sarcastic class.

Overall, there are small differences observed in the F-score for the non-sarcastic class, apart from the individual L_{tr} , L_g lexicon-based representations, which perform the worst for almost all algorithms. The latter is also the case for the sarcastic class, in which the lexicon-based representations perform very poorly. On the one hand, this might imply that our lexicons are unable to deal with sarcasm. On the other hand, given that sarcasm detection is a rather context-dependent task, this might also mean that our lexicons' contribution to this task should be evaluated in a cross-domain manner, similar to Task 1. Nevertheless, both L_g and L_{tw} confidently outperform L_{tr} , whereas merging them into L yields consistently better results than the individual L_g and L_{tw} for all algorithms and classes. Word embeddings, on the other hand, outperform all lexicon-based approaches in almost all cases and form a competitive feature source against n-grams for this task.

Table 4.7: Pearson correlation for the Emotion Prediction task (Task 2), using 5-fold cross validation.

emotion	algorithm	baselines		our resources				combinations			
		N	L _{tr}	L _g	L _{tw}	L	E	NL	NE	LE	NLE
fear	MC _{avg}	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
	MC _{dist}	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
	LASSO	.200	-.020	.043	.119	.092	.148	.213	.243	.162	.226
	RFR	.192	.007	.086	.214	.203	.188	.266	.222	.192	.225
	SVR	.197	.022	.146	.210	.196	.276	.135	.239	.278	.240
	average	.196	.003	.092	.181	.164	.204	.205	.235	.211	.230
happy	MC _{avg}	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
	MC _{dist}	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
	LASSO	.345	.099	.276	.283	.324	.353	.341	.360	.353	.364
	RFR	.370	.162	.343	.429	.446	.499	.458	.498	.502	.501
	SVR	.388	.158	.287	.471	.462	.501	.409	.468	.495	.463
	average	.368	.140	.302	.394	.411	.451	.403	.442	.450	.443
sad	MC _{avg}	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
	MC _{dist}	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
	LASSO	.311	.071	.184	.218	.213	.267	.322	.355	.263	.361
	RFR	.357	.061	.226	.346	.376	.452	.400	.453	.453	.453
	SVR	.358	.094	.161	.346	.327	.443	.249	.409	.428	.395
	average	.342	.075	.190	.303	.305	.387	.324	.406	.381	.403
surprise	MC _{avg}	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
	MC _{dist}	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
	LASSO	.265	.067	.084	.277	.258	.259	.272	.376	.269	.385
	RFR	.417	.073	.226	.442	.465	.513	.480	.519	.517	.521
	SVR	.370	.031	.143	.399	.388	.449	.364	.415	.482	.451
	average	.351	.057	.151	.373	.370	.407	.372	.437	.423	.452

Table 4.8: Cross-emotion results for Task 2.

emotion	baselines		our resources				combinations			
	N	L _{tr}	L _g	L _{tw}	L	E	NL	NE	LE	NLE
MSE-average	1.76	1.83	1.77	1.53	1.54	1.73	1.73	1.59	1.72	1.57
MSE-best	1.55	1.72	1.65	1.45	1.44	1.35	1.41	1.35	1.34	1.35
ρ -average	.314	.069	.184	.313	.313	.362	.326	.380	.366	.382
ρ -best	.341	.088	.235	.368	.377	.436	.401	.428	.438	.429

The comparison between the rest of the resources shows that there is a small improvement when combining different feature sets over n-grams or word embeddings. Overall, the best macro-average score is achieved by SVM, when trained on word embeddings and n-gram features, outperforming the best n-gram-based model by almost 1%. While this improvement is relatively small, it is worth noting that those results are achieved using 10-fold cross-validation on the same dataset and not in a different domain, in which the n-grams tend to perform a lot worse in sentiment-related tasks, as demonstrated in Table 4.5. Cross-domain sarcasm detection is a challenging direction for future work.

Table 4.9: F-score on the Sarcasm Detection Task.

class	model	baselines		our resources				combinations			
		N	L _{tr}	L _g	L _{tw}	L	E	NL	NE	LE	NLE
Non-sarcastic	MC	88.47	88.47	88.47	88.47	88.47	88.47	88.47	88.47	88.47	88.47
	LR	92.75	88.48	88.76	91.00	91.21	90.87	92.79	91.97	91.33	91.85
	RF	92.93	88.51	88.73	90.11	90.42	93.01	91.59	92.65	92.96	92.81
	SVM	92.34	88.49	88.59	87.20	87.22	92.64	92.30	93.46	92.28	93.40
Sarcastic	MC	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	LR	70.94	0.77	22.43	57.70	59.05	64.52	71.37	67.93	66.21	67.92
	RF	71.61	12.11	33.43	50.72	52.10	68.50	59.72	65.53	68.84	67.04
	SVM	72.32	11.79	21.70	33.99	39.31	68.63	71.50	73.14	68.50	73.10
Macro-average	MC	44.23	44.23	44.23	44.23	44.23	44.23	44.23	44.23	44.23	44.23
	LR	81.85	44.63	55.59	74.35	75.13	77.69	82.08	79.95	78.77	79.88
	RF	82.27	50.31	61.08	70.41	71.26	80.76	75.65	79.09	80.90	79.93
	SVM	82.33	50.14	55.14	60.60	63.26	80.64	81.90	83.30	80.39	83.25

4.4.4 Key Findings

Our results demonstrate the effectiveness of our resources in all studied tasks. While the accuracy that is expected using our resources in a particular task may vary (i.e., due to the limited resources in the Greek language, we were restricted to five datasets overall), the boost in performance when employing our lexicons and embeddings are consistent in all cases. Overall, our main findings with respect to the effectiveness of our resources in the three studied tasks are summarized as follows:

1. In the intra-domain sentiment analysis and sarcasm detection tasks, the n-gram representation is hard to beat. This is expected, since n-grams form a competitive representation due to their nature of capturing word-to-class associations within a single domain, under the assumption that such information (i.e., domain-specific annotations) are available. Nevertheless, by using strictly our resources or our resources alongside the n-gram feature set for the sentiment analysis task, we obtain an average (across-datasets) relative improvement of 2.7%–5.6%, depending on the algorithm used. For sarcasm detection, the differences in F-score for our resources in comparison with the n-gram baseline are minor, primarily due to the context-dependent nature of the task, which is captured effectively by the n-grams.
2. On the contrary to the above finding, in the emotion detection task, the

n-gram representation is performing quite poorly, achieving the lowest correlation and highest error rates when compared to our lexicons and word embeddings. We achieve 9.5% improvement in Pearson correlation and 0.2 error reduction rates, by using only our word embedding representation, whereas the addition of other features yields only minor differences in terms of accuracy. The reason for this effect is that the emotion intensity task was not studied on a single domain; hence, our word embeddings, which are trained over a large and generic corpus, form a more appropriate feature extraction method for this type of task.

3. The major advantage of our resources is highlighted in the cross-domain sentiment analysis task, which is the task that motivates the creation of such resources. Given that it is impossible to have annotated datasets for all domains and purposes, creating lexicons and resources that can be used in a new domain is of crucial importance in sentiment analysis. Here we demonstrated that we achieve a clear improvement in accuracy (24.9% relative improvement on average, across the three algorithms in Table 4.5) over the best n-gram model. Importantly, a similar improvement (22.7% across the three algorithms) results from using features derived strictly from our resources, again improving the computational load of any algorithm.
4. Finally, in all tasks, we observe that our GrAFS lexicon consistently outperforms the translated one. However, our Twitter-based lexicons (KBL, EBL) form much better feature extraction resources for all tasks, clearly demonstrating the importance of building resources for handling user-generated content, which is not captured by our expanded GrAFS lexicon. Nevertheless, we plan to investigate whether the same conclusion holds when dealing with more well-formed documents, such as news articles.

4.5 Summary and Conclusion

In this chapter we presented the generation and evaluation of various rich resources for sentiment-related analysis for an under-resourced language (i.e., the Greek language). We have evaluated our resources in-depth, with promising results. Importantly, our evaluations moved beyond the popular sentiment analysis task, demonstrating the effectiveness of our resources in multiple related tasks, including *emotion* and *sarcasm detection*. By releasing our resources, we aspire to encourage and support research on sentiment-related tasks in the Greek language. Having set the basis for the user-agnostic, document-level analysis in social media, in the next chapters we move into the temporal modelling of social media streams of data and study their effectiveness at capturing real-world indices, both at the macro- as well as the micro-level.

CHAPTER 5

Macro-level Modelling Using Social Media

RQ1. *Can we use data streams from social media in order to nowcast real-world indices on the macro-level?*

The current chapter presents the first part of nowcasting real-world indices using social media, effectively addressing RQ1 in the macro-level setting. In specific, here we focus on the political domain, aiming to use large data streams from social media in order to predict the election results in three different electoral races. Modelling our problem as a time-series task, we demonstrate that social media can effectively be used alongside traditional polling methods to provide better estimates of the current state of macro-level political indices (i.e., voting shares of the major political parties) and, thus, of the final election results. In order to provide evidence of the real-world applicability of our method, we are the first, to the very best of our knowledge, to publish accurate model estimates for an electoral race *before* the end of the elections, while we follow the exact same methodology for the other two electoral races, achieving similar performance. We outperform various past works, state-of-the-art methods and election prediction services, demonstrating the importance of the features derived from social media for the task. Finally, while the presented approach is tested on a single domain, its adoption in other macro-level tasks and domains is non-trivial¹.

¹The current chapter is based on [243].

5.1 Introduction

Twitter has seen an amplified overall interest over the latest years, recording about 500 million short messages sent per day². Hence, it is not surprising that it is increasingly exploited for various research tasks, including modelling and predicting users' behaviour at the macro-level.

The current chapter focuses on exploiting content derived from large Twitter data streams for the task of predicting macro-level indicators. While we focus on a particular domain (i.e., the political domain), our approach can easily be adjusted to fit other macro-level purposes. In particular, in this chapter we focus on the task of predicting the 2014 European Union (EU) Election results in Germany, the Netherlands and Greece. While several works have been conducted on the same domain, many of them relied strictly on Twitter data and have been proven ineffective when tested in different elections. Furthermore, most of the past works have published their results after the elections; others raised questions on the benefit of using Twitter data for this task (see section 3.3.1).

In the modelling presented in this chapter, we treat the users' voting intentions as time-variant features. Instead of trying to predict every user's vote (as in Chapter 6 that follows up next), we treat Twitter political discussions as a general macro-level index that varies with time; we define several Twitter-based features and fit them in time-series models, using opinion polls as our ground-truth. In this way, we combine the Twitter-based time-series with the poll-based ones. We test three different forecasting algorithms using three different sets of features; we contrast our results with several popular methods, achieving lower error rates even compared to prediction websites and polls. Furthermore, working on different elections at the same time, we demonstrate the portability of our approach. Most importantly though, we show that by using the proposed Twitter-based features, all tested algorithms get a significant boost in accuracy, compared to when using only poll-based features. Last but not least, we are among the first to have published our predictions before the announcement of

²<https://about.twitter.com/company>

the Exit Polls for one country³, preventing any bias towards them, while we follow the exact same methodology for the other two countries.

5.2 Background: The EU Elections

The EU Parliament elections are held every five years among the EU member states. Elections take place almost simultaneously across Europe and people vote for the national parties of their countries. The 2014 EU elections were judged as extremely important, in light of the economic crisis and the rise of Euroscepticism. Due to the nature of these elections, it is difficult to predict the results at a pan-European level without taking into account the important demographic and political differences between the EU members. Thus, here we focus on three different countries, transferring the problem to a national level. The elections were held on May, 22nd for the Netherlands and on May, 25th for Germany and Greece. There were 10 main political parties contesting in the Netherlands, 6 in Germany and 8 in Greece.

5.3 Methodology

Approaching our problem as a multivariate time-series forecasting task for each country separately, we create time-series of 11 Twitter- and one poll-based features for every party (sections 5.3.2–5.3.3). An example is the number of tweets mentioning a certain party on a specific day (Twitter-based) and the percentage for that party reported on a poll that was conducted on that day (poll-based). Next, we provide all these features as input to different forecasting algorithms for predicting the voting share of every party separately (section 5.3.4).

³Our estimates of the final results for Greece were published at: <http://www.socialsensor.eu/news/133-sensing-social-media-to-predict-eu-elections>

5.3.1 Data Aggregation

We started aggregating data published on Twitter and various opinion polls on a per-country basis from April, 6th until two days before the elections (20/5 for the Netherlands and 23/5 for Germany and Greece), leaving one day to conduct our processing. Using the public Twitter Streaming API⁴, we aggregated tweets written in the respective language that contained a party’s name, its abbreviation, its Twitter account name and some possible misspellings (e.g., “grunen” instead of “grünen”). We excluded several ambiguous keywords in order to reduce noise (e.g., the abbreviation of the Dutch party “GL” may stand for “good luck”), which might slightly affect the replication of naïve counting-based methods⁵.

5.3.2 Modelling

Twitter Features We extract several Twitter-based features potentially disclosing the users’ voting intentions at the large-scale. These features were based on past works, showing that the counts of a political party on Twitter and the expressed sentiment towards it are – to some extent – related with its voting share in the elections. However, instead of relying strictly on counting-based methods, we incorporate daily features into time-series, in order to correlate them to the opinion polls (discussed next).

Working on every country separately, we first assigned equal weights to all parties mentioned in a tweet, so that they sum up to one. Let $t_d(p)$ denote the (weighted) number of tweets that mention party p on day d and $tpos_d(p)$ ($tneg_d(p)$) the corresponding number of tweets containing positive (negative) content. Similarly, let $u_d(p)$ denote the number of users mentioning party p on day d and $upos_d(p)$ ($uneg_d(p)$) the number of users that have published a tweet with positive (negative) content about party p on that day. We constructed the

⁴<https://dev.twitter.com/>

⁵The complete list of keywords that was used for Twitter Streaming API is provided in Appendix B. The aggregated tweet ids are available at http://socialsensor.eu/images/files/eu2014_prediction_sup_material.zip.

following features, for every day:

- $numTweets_d = \sum_i t_d(i)$
- $pctTweets_d(p) = \frac{t_d(p)}{\sum_i t_d(i)}$
- $pctTPos_d(p) = \frac{tpos_d(p)}{t_d(p)}$
- $pctTNeg_d(p) = \frac{tneg_d(p)}{t_d(p)}$
- $pctTPosShare_d(p) = \frac{tpos_d(p)}{\sum_i tpos_d(i)}$
- $pctTNegShare_d(p) = \frac{tneg_d(p)}{\sum_i tneg_d(i)}$
- $pctUsers_d(p) = \frac{u_d(p)}{\sum_i u_d(i)}$
- $pctUPos_d(p) = \frac{upos_d(p)}{u_d(p)}$
- $pctUNeg_d(p) = \frac{uneg_d(p)}{u_d(p)}$
- $pctTotalUsers_d(p) = \frac{\sum_{x=0}^d u_x(p)}{\sum_{x=0}^d u_x(i)}$

Here, $pctTotalUsers_d(p)$ refers to the *distinct* number of the users that have mentioned p , divided by the total number of users up to day d . We also added the average sentiment value ($avgSentiment_d$) as an eleventh feature (notice that $numTweets_d$ and $avgSentiment_d$ were the same for all parties within a country). Finally, we used a 7-day Moving Average (MA) filter for all features (except $pctTotalUsers_d(p)$) in order to normalise their values, as in [173]. These 11 values for every party were used as our Twitter-based features and were provided as input to our algorithms, along with the opinion poll ones, which are presented next.

Opinion Polls Since there is not a complete polling aggregation service, we had to find different polls manually. Once aggregated, we removed all poll values from “small” parties (not appearing in all polls) and added their voting share to the “Others” bucket, since we were only interested in the main parties of each country; then, we distributed proportionally to all parties (including

“Others”) the voting share of all “undecided” voters. In this way we managed to have consistent polls, adjusting their reports to include only the main political parties of each country, along with the “Others”.

While creating time-series of Twitter features without missing values was a straightforward process, this was not the case for the polls. A poll is usually conducted over two to three days; we treated the adjusted results as the actual voting shares each party would have received if the elections were held on any of these days. If two or more polls were held on the same day, we considered the voting share of each party as the weighted average value, using the sample size of every poll as the weight and making sure that all voting shares sum up to 100. We then filled all days without polling data by using linear interpolation. Finally, for the days after the last poll, we replicated the last poll-based value for every party in order to set the prediction horizon to 1 for our predictive algorithms, for consistency between the different countries. There was only one such day (that is, the last day) for Germany and the Netherlands.

5.3.3 Sentiment Analysis

Several Twitter-based features were sentiment-related; hence, we needed to assign a sentiment value to each tweet before proceeding. Sentiment analysis is usually performed in a supervised fashion (see previous chapter). However, past works have revealed the domain-dependent nature of such classifiers [208]. The integration of Part-of-Speech (POS) tags is also beneficial, but there does not exist a reliable, free-to-use POS tagger for the three languages. Finally, the different performance of the sentiment classifiers across the three languages could be another barrier in creating approaches that can generalise across different cases. Given these constraints, we decided to adopt a lexicon-based approach, in order to create a generic method that could be applied in different cases. Since we had only generated such resources for the Greek language (see previous chapter), we refrained from employing it here, for consistency purposes across the different languages. While such lexicon-based approaches perform

only slightly better than a random classifier [150], we were only interested in the daily differences of the expressed sentiment; thus, given that we have enough data on every day, even a slightly better than the random classifier method could fit our goals [173].

Due to the lack of a sentiment lexicon for different languages, we translated three English lexicons using Google Translate⁶. The lexicons we employed were the following:

SentiWordNet⁷ [71] contains information about 150,000 synsets, with a double value indicating their polarity;

Opinion Lexicon⁸ [100] contains two lists (positive, negative) of about 6,800 polarized terms;

Subjectivity Lexicon serves as part of the Opinion Finder⁹ [258] and contains about 8,000 terms along with their Part-of-Speech (POS), subjectivity – strong or weak – and polarity indication.

We assigned the values of 1 and -1 for the positive and negative terms in the Opinion Lexicon respectively; for the Subjectivity Lexicon, we used four values (-1, -0.5, 0.5, 1) to represent every subjective word based on subjectivity ($|0.5|$ for weak, $|1|$ for strong) and polarity; for SentiWordNet, we kept the exact values of every synset. We removed all terms that were not a single word, due to the inaccuracy observed in those translations. If the same word appeared in different lexicons, we considered the average as its sentiment value, resulting into 14,060/19,357 German, 13,838/18,993 Dutch and 13,582/18,356 Greek positive/negative terms. In order to detect a tweet’s sentiment, we used a naïve sum-of-weights method on its keywords, according to the respective lexicon, and assigned the majority class label (positive/negative) to it.

⁶<https://translate.google.com/>

⁹<http://mpqa.cs.pitt.edu/opinionfinder/>

5.3.4 Algorithms

We tested three different algorithms on each political party separately, using only this specific party’s features (11 Twitter- and one poll-based) as input. These algorithms were Linear Regression, Gaussian Process and Sequential Minimal Optimization for Regression, all implemented using Weka¹⁰ with the default settings¹¹. Since it was difficult to evaluate each algorithm before the elections, we initially decided to empirically apply a seven-day training window for every algorithm and considered the average predicted percentage for every party as our final estimate.

5.4 Data

5.4.1 Twitter

We aggregated 361,713 tweets from 74,776 users in Germany, 452,348 from 74,469 users in the Netherlands and 263,465 from 19,789 users in Greece (see Figure 5.1). Our findings on the average sentiment value reveal that negative opinions dominate in political discussions (-0.54 for Germany, -1.09 for the Netherlands and -0.29 for Greece). Figure 5.1 shows that there were far more tweets published in the week before the elections, whereas a slight decrease is noticed in the Easter week (13–20/4). Still, due to the restrictions of the Twitter Streaming API (it returns no more than 1% of all public tweets), we could have missed some data. Research has shown that the increase of global awareness on a topic, or the sudden decrease in tweets on a day could result into a decrease of the coverage of the Streaming API and, consequently, lead to a noisy bias [169]. However, since the total amount of aggregated data is fairly moderate, our data loss (if any) is probably negligible. Moreover, as we are only interested in time-series modelling, this should not affect our process.

¹⁰<http://www.cs.waikato.ac.nz/ml/weka>

¹¹Our released results included a fourth algorithm (Support Vector Regression); due to its poor performance in Greece, we did not test it on the other countries.

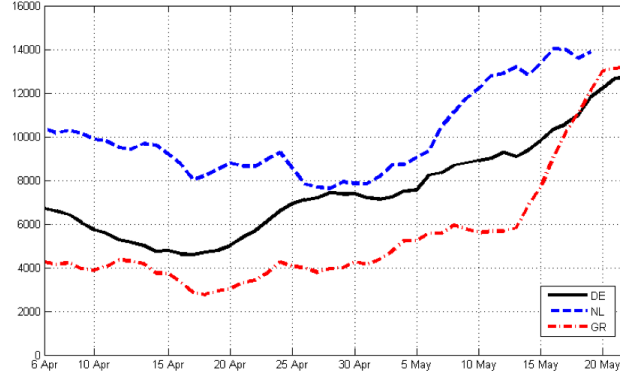


Figure 5.1: Number of political tweets aggregated per day in the three electoral races, after a 7-day MA filter.

5.4.2 Opinion Polls

In total, we used 26 different opinion polls from 11 diverse sources in Greece, 9 polls from 4 sources in Germany and 13 polls from 3 sources in the Netherlands. More specifically, we used all the polls published in MetaPolls¹²; further resources used were <http://www.wahlrecht.de/> for Germany, <http://www.3comma14.gr/> for Greece and polls from Ipsos¹³, TNS Nipo¹⁴ and Peil¹⁵ for the Netherlands.

Table 5.1 shows the variance of every party’s voting share based on all collected poll results, after our pre-processing (for the Netherlands the “Others” category was missing in most polls and thus not included in our analysis). The voting shares of the German parties are rather stable, unlike the percentages reported for the Dutch and the Greek parties, reflecting the differences of people’s voting intentions through time. Not surprisingly, since polls were part of our training process, we achieved lower error rates in Germany than the other countries (see next section), for which the prediction task is more challenging.

¹²<http://metapolls.net/>

¹³<http://www.ipsos-nederland.nl/>

¹⁴<http://www.tns-nipo.com/>

¹⁵<https://www.noties.nl/peil.nl/>

Germany		The Netherlands		Greece	
CDU/CSU	1.15	PVV	1.80	ND	3.39
SPD	0.99	VVD	3.88	SYRIZA	4.29
Grünen	1.00	D66	1.41	XA	2.36
Linke	0.63	CDA	3.38	Potami	5.38
AfD	0.25	PvdA	1.85	KKE	0.54
FDP	0.25	SP	2.00	Elia	1.49
Others	1.00	CU/SGP	0.64	ANEL	0.62
		GL	0.46	DIMAR	0.36
		50+	0.24	Others	1.95
		PvdD	0.40		
Average	0.75	Average	1.61	Average	2.51

Table 5.1: Variance of reported shares in the processed polls.

5.5 Results

In the current section we present the results obtained from our method (denoted as “Twitter-Poll-Based”, “**TPB**”), along with several other competing methods. These were a combination of established naïve methods, past works, commercial resources and our method when leaving some features out:

CB1 The Count-Based method by Tumasjan et al. [247].

CB2 A similar naïve method [216], applied by keeping the tweets that mention only one party and then the first tweet of every user. At the final stage, voting shares are given to the parties as in **CB1**. In both “CB” cases, since we did not have data for the last day before the elections, we worked on the last seven days that we had data for (for the week ending two days before the elections).

SB This is a replication of the work by Sang and Bos [216]. We train on all polls before the last week. For sentiment analysis, we use our own naïve dictionary-based method. In the original paper, sentiment values are given to the parties after manual annotation of some tweets. Nevertheless, these sentiment values are then adjusted to the “population weights”, so our sentiment analysis choice should not affect the results.

Polls The average of the polls conducted during the last processing week; due

to the different companies publishing their poll results at the same time, we provide the average of their reports. There was one poll in Germany, two in the Netherlands and seven in Greece.

MP This baseline refers to the predictions of MetaPolls. To the best of our knowledge, this is one of the only two websites providing predictions for all EU countries. MetaPolls provide their voting estimates for every party in a range of values; we considered the average value of this range for every party as the predicted percentage, making sure that the values sum up to 100.

PW Similarly, PollWatch¹⁶ is the official prediction website that is powered by VoteWatch Europe and Burson-Marsteller/Europe Decides.

PB In order to evaluate the use of our Twitter features, we compare against using only polling data as features. Hence, this Poll-Based method averages our three algorithms, fed only with poll-based data.

CPB Similarly, the “Counting-Poll-Based” method was used to evaluate the performance of our sentiment analysis features. Hence, its features include the polling data points along with all the non-sentiment Twitter features (*numTweets*, *pctTweets*, *pctUsers*, *pctUsersTotal*).

In order to evaluate both the voting share predictions of the competing methods and the ranking of the parties, we selected the standard Mean Absolute Error (MAE), the Mean Squared Error (MSE) and the Tau Kendall Coefficient (τ_a) as our evaluation metrics. Tables 5.2-5.4 present the results for the electoral races in every country separately. Table 5.5 presents the average per-country values of our evaluation metrics.

As expected, naïve methods perform the worst. While in [216] *CB2* provided a boost in accuracy compared to *CB1* against polls, our findings show that this accuracy drops against the actual results. Surprisingly though, these

¹⁶<http://www.electio2014.eu/>

Party	Result	CB1	CB2	SB	Polls	MP	PW	PB	CPB	TPB
CDU/CSU	35.30	18.38	17.98	33.36	37.50	37.76	37.70	37.51	37.93	37.09
SPD	27.30	20.52	17.78	31.74	26.50	26.53	27.00	26.47	26.35	26.87
Grünen	10.70	9.41	10.25	6.67	10.00	10.07	10.70	9.76	9.33	9.82
Linke	7.40	10.25	8.64	7.64	7.50	8.28	8.30	7.22	7.66	7.24
AfD	7.10	23.55	17.91	9.92	7.00	6.58	6.30	7.10	7.08	7.56
FDP	3.40	9.09	18.63	1.87	3.50	3.39	3.00	3.84	3.47	3.38
Others	8.80	–	–	–	8.00	7.38	7.00	8.10	8.18	8.04
MAE	0.00	8.33	9.10	2.50	0.69	0.69	0.94	0.76	0.85	0.64
MSE	0.00	107.49	123.53	8.36	0.95	0.95	1.53	1.02	1.45	0.71
Tau	1.00	0.20	-0.07	0.60	1.00	0.90	1.00	1.00	1.00	0.90

Table 5.2: Comparison between the performance of our approach against various baselines (Germany).

Party	Result	CB1	CB2	SB	Polls	MP	PW	PB	CPB	TPB
D66	15.87	14.79	13.58	24.96	17.49	17.68	18.53	16.49	16.22	15.72
CDA	15.56	8.73	9.46	13.13	11.84	12.63	11.46	13.33	13.70	13.16
PVV	13.65	14.94	10.80	15.40	16.28	13.64	14.23	15.66	16.09	16.70
VVD	12.32	12.25	11.87	14.58	16.26	13.13	13.92	15.51	15.09	15.51
SP	9.84	17.71	21.30	8.84	12.97	12.32	11.46	13.56	13.28	13.33
PvdA	9.64	13.10	12.35	6.59	7.64	10.00	10.33	6.88	7.28	7.11
CU-SGP	7.86	5.57	7.84	4.22	7.21	9.60	9.32	7.90	7.49	7.77
GL	7.16	7.85	6.96	5.87	4.47	5.25	5.73	4.77	4.91	4.77
PvdD	4.32	4.20	4.39	4.59	3.24	2.22	1.44	3.32	3.40	3.40
50+	3.78	0.86	1.45	1.80	2.60	3.54	3.58	2.58	2.54	2.52
MAE	0.00	2.66	2.85	2.68	2.26	1.44	1.72	1.91	1.80	1.94
MSE	0.00	13.76	19.50	12.59	6.28	2.99	4.24	4.91	4.22	5.19
Tau	1.00	0.56	1.45	0.82	0.87	0.87	0.85	0.82	0.87	0.78

Table 5.3: Comparison between the performance of our approach against various baselines (The Netherlands).

Party	Result	CB1	CB2	SB	Polls	MP	PW	PB	CPB	TPB
SYRIZA	26.60	22.55	21.16	23.77	28.60	29.00	29.60	27.72	26.48	27.25
ND	22.71	18.60	15.58	19.77	25.32	25.50	26.00	25.81	23.89	24.67
XA	9.38	14.92	22.71	13.47	9.45	9.40	8.00	10.02	9.45	9.06
Elia	8.02	3.99	7.23	6.57	7.13	7.30	6.50	7.40	8.31	8.10
Potami	6.61	4.39	6.24	3.46	7.73	7.70	8.00	6.56	10.73	8.07
KKE	6.07	8.17	7.19	10.02	6.50	6.10	6.00	5.97	6.30	6.16
ANEL	3.47	9.61	2.83	5.38	4.09	4.00	5.10	4.16	3.63	4.26
DIMAR	1.21	1.85	1.13	1.62	2.32	2.40	3.20	2.61	1.90	2.72
Others	15.93	–	–	–	8.85	8.60	7.60	9.75	9.31	9.71
MAE	0.00	3.60	3.61	2.59	1.77	1.79	2.51	1.55	1.50	1.45
MSE	0.00	15.96	32.56	8.10	7.20	7.85	11.33	5.82	6.98	5.34
Tau	1.00	0.57	0.79	0.79	0.89	0.89	0.82	0.94	0.78	1.00

Table 5.4: Comparison between the performance of our approach against various baselines (Greece).

approaches are the most successful for certain parties – *CB2* is the best method for 4/26 parties; despite that, it is the worst method overall. *SB* fails to perform competitively compared to other approaches. Whilst this might be influenced by the different sentiment analysis method used, it mainly suggests the impor-

Method	MAE	MSE	Tau
CB1	4.87	45.74	0.44
CB2	5.19	58.53	0.42
SB	2.59	9.68	0.74
Polls	1.57	4.81	0.92
MP	1.39	4.10	0.89
PW	1.73	5.70	0.89
PB	1.40	3.91	0.92
CPB	1.38	4.21	0.88
TPB	1.35	3.75	0.89

Table 5.5: Comparison between the performance of our approach against various baselines, macro-averaged across the three electoral races.

tance of computing macro-level indicators of people’s voting intentions at the large-scale as time-variant features, instead of as static values.

Polls error values vary a lot among the three different countries. In Germany, *Polls* was the second best predictor for the final result, in terms of MAE. However, in both Greece and the Netherlands, it performed relatively poorly, compared to other poll-based methods. This is an interesting point: although our models (*TPB*, *PB*, *CPB*) used polls for training, they manage to outperform *Polls* in both error metrics, by using knowledge from the past. Given that every poll has a standard error (usually around 3%) along with a certain number of undecided voters, treating polls (next to other features) as time-series seems a better practice. Overall, only two (out of ten) polls conducted during the last week achieved better results in MAE than our *TPB* approach (one in Greece, with MAE 1.35, and one in the Netherlands, with MAE 1.78). Moreover, *Polls* have the second highest value; however, the differences among most models are minor. From the two prediction websites, *MP* outperformed *PW* by a margin of 0.33 in MAE. The most likely reason for this big difference is that *MP* released their predictions one day before the elections, whereas *PW* published them on 20/5 for all countries.

Overall, our *TPB* model performed the best in both error rate terms. However, it did not perform equally well in terms of correct ranking of the parties, following by 0.03 the best competing model (*PB*) in τ_a . One possible expla-

nation of this effect is that, as we were not interested in correctly ranking the political parties, but instead predicted their voting shares individually, only the features related to an individual party were used for the prediction for this party. Using features from different parties in order to predict each party’s voting share is a challenging task for future research.

From the three algorithms used in *TPB*, Gaussian Process achieved the lowest MAE (1.31), followed by Sequential Minimal Optimization (1.35) and Linear Regression (1.42). This means that Gaussian Process performed better than our “averaging” *TPB* model. However, since we did not know how reliable the polls were, we did not have a guaranteed ground-truth before the elections and the “averaging” method seemed a safer choice.

Importantly, the comparisons between *TPB* and *PB* (and between *TPB* and *CPB*, respectively) show that our Twitter features were beneficial. However, the differences between error rates are rather small. Furthermore, in the case of the Netherlands, *PB* and *CPB* achieved better results. So, whilst our approach achieved the best results overall, the exact contribution of Twitter and sentiment features needs to be further explored, as follows.

5.6 Post-hoc Analysis and Discussion

Recall that all of our models (*TPB*, *PB*, *CPB*) were based on a 7-day training window and the average of the predictions by Linear Regression (LR), Gaussian Process (GP) and Sequential Minimal Optimization for Regression (SMO) were reported in our results. Both of these decisions (window size, averaging) were made empirically, since we did not know the actual results. In order to better compare these models, we applied the same algorithms trained on five different window sizes (starting from one-week with weekly increases of training size up to five-weeks). In this section, we also provide the results obtained from each individual algorithm (LR, GP, SMO) for every model.

Figure 5.2 presents the MAE values for all countries and for all algorithms

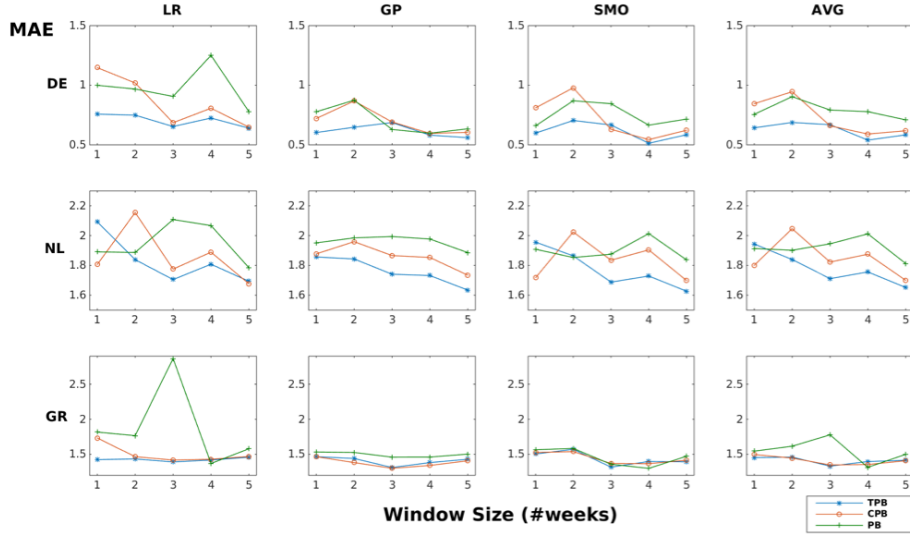


Figure 5.2: MAE per training window size for different algorithms and countries.

(including our “averaging” – “AVG” – approach) when trained on different sets of features (leading to *TPB*, *PB*, *CPB* models) and on different time windows. In most cases, the error drops when we use our complete *TPB* model’s features; this holds in 71% of the total cases for the individual algorithms (12/15 for LR and 10/15 for GP and SMO) and 67% for the “averaging” method (10/15). We also notice that, in most cases, the errors follow a downwards trend as the training window size increases. Hence, training over a wider period seems beneficial, although finding the optimal window remains a challenging task for future research.

On a per-window size, cross-country average, LR performs consistently worse than GP and SMO in all feature models (the only exception being for *TPB* in the two-week training window), indicating that it was a poor choice to include it in our models. The differences between GP and SMO are minor. Using our *TPB* model’s features, GP achieves more stable MAE values across different window sizes, ranging on a cross-country average from 1.21 (5-week) to 1.31 (1-week); however, for the same features, the best cross-country performance in MAE terms is achieved by SMO (1.20, for the 5-week training window size).

Despite the better overall performance of all algorithms comprising our *TPB* model when compared to *PB* and *CPB* (see Figure 5.2), we need to further test for significant differences between the different models. Thus, we applied the non-parametric Wilcoxon test (two-tailed) to all three algorithms, as well as our “averaging” initial approach, using the MAEs and MSEs obtained by every algorithm on every country and training window size.

Comparing *TPB* and *PB* revealed that for all three algorithms as well as the “averaging” approach, there exist significant differences in MAE for the level of .05. The same test of the respective MSEs revealed significant differences for the .05 level for LR, GP and the “averaging” approach, but not for SMO ($p = .057$). Comparing *TPB* with the *CPB* MAE rates revealed significant differences for the .05 level for all methods except SMO; for the same level, the differences for all algorithms and our “averaging” method were significant when applied to the MSE rates. These results highlight the importance of all our Twitter with sentiment-based features: by incorporating them into our prediction models we achieve better error rates and, in most cases, significant differences to features using only polls or polls with count-based (no-sentiment) features. Given that our work was unbiased towards the election results, these conclusions provide highly supportive evidence on the potential of using social media data for the election prediction task.

5.7 Summary and Conclusion

The current chapter focused on nowcasting macro-level political indicators using social media, with an application on the domain of predicting the 2014 EU elections for three countries. Working on time-series modelling, we extracted real-world, macro-level indicators (polling data) and several features from political tweets and trained different algorithms on them. Our results demonstrate the appropriateness of our approach in error rate terms, achieving better results than real-world macro-level indicators (i.e., opinion polls), predic-

tion websites and replication of previous works. Most importantly though, we demonstrated that by incorporating certain features derived from social media into those derived from real-world macro-level indicators (i.e., poll-based features), we increase accuracy in a statistically significant way, whilst the same conclusions were reached when sentiment-related features are used compared to strictly counting-based ones. Finally, while the focus on this chapter was on the political domain, the methodology can easily be adjusted to model other social or urban macro-level indicators in a time-sensitive manner, using social media.

Part III

Micro-level Modelling Using Social Media and Smart Devices

CHAPTER 6

Micro-level Modelling Using Social Media

RQ2. *Can we use data streams from a specific group of social media users in order to nowcast their real-world indices on the micro-level?*

Chapter 5 presented an approach on mining large streams of social media data to nowcast real-world (political) indices on the macro-level. Such approaches fail to capture an individual user's index, which is often of interest, and they also demand ground-truth scores stemming from real-world indices, thus failing to create independent alternative sensors of the current urban/social state.

The current chapter zooms into the micro-level modelling of social media users in a longitudinal manner, within the political domain. In particular, we focus on the case of the Greek bailout referendum (2015), which was suddenly announced in June, 27th and was held eight days later, aiming to predict the voting intention of 2,197 users on a longitudinal basis, using data derived from their social media. We extract temporally sensitive (a) linguistic and (b) network features to represent the users on a daily basis, modelling them through convolution kernels which are combined under a multiple kernel learning approach. Our results under a real-time simulation framework demonstrate the effectiveness and robustness of our approach against competitive baselines, achieving a significant 20% increase in F-score compared to solely text-based models. Finally, we provide qualitative insights on the importance of temporal modelling at the micro-level for our task, through multiple convolution kernels¹.

¹The current chapter is based on [240] – accepted for publication in the 2018 ACM Conference on Information and Knowledge Management.

6.1 Introduction

Predicting user voting stance and final results in elections using social media content is an important area of research in social media analysis [150, 78] with applications in online political campaigning and advertising [99, 42]. It also provides political scientists with tools for qualitative analysis of electoral behaviour on a large scale [4]. Previous approaches mainly focus on predicting national general elections at the macro-level (as presented in Chapter 5), which are regularly scheduled and where data of past results and opinion polls are available [126, 243]. However, the task of predicting the result of an electoral race lacks of robust evaluation, since model performance is assessed based on a few instances (typically, the voting share of 2–10 political parties). Furthermore, there is no evidence of how such models would work during a *sudden* and *major* political event under time-constrained circumstances, where the time-series modelling presented in Chapter 5 cannot be applied due to the short duration of the pre-electoral race. That forms a more challenging task compared to general elections, due to its spontaneous nature. Building robust methods for voting intention of social media users under such circumstances is important for political campaign strategists and decision makers.

The current chapter focuses on nowcasting the voting intention of Twitter users in the 2015 Greek bailout referendum that was announced in June, 27th 2015 and was held eight days later. Acknowledging the demographic bias on Twitter [156], we do not attempt to predict the actual result. Instead, we define a time-sensitive binary classification task where the aim is to classify a user’s voting intention (YES/NO) at different time points during the entire pre-electoral period.

For this purpose, we collect a large stream of tweets in Greek and manually annotate a set of users for testing. We also collect a set of users for training via distant supervision. We predict the voting intention of the test users during the eight-day period until the day of the referendum with a multiple convolution

kernel learning model. The latter allows us to leverage both temporally sensitive textual and network information. Collecting all the available tweets written in Greek², enables us to study user language use and network dynamics in a complete way. We demonstrate the effectiveness and robustness of our approach, achieving a significant 20% increase in F-score against competitive text-based baselines. We also show the importance of combining text and network information for inferring users' voting intention.

In this chapter, we make the following contributions:

- We present the first systematic study on nowcasting the voting intention of Twitter users at the micro-level during a sudden and major political event, under a real-world setting.
- We propose a novel Multiple Convolution Kernel Learning (MCKL) approach, operating on temporally sensitive linguistic and network-related information about the users, aiming to nowcast their voting intention.
- We demonstrate that network and language information are complementary, by combining them with multiple convolution kernels.
- We highlight the importance of the temporal modelling of text for capturing the voting intention of Twitter users.
- We provide qualitative insights on the political discourse and user behaviour during this major political crisis, reasoning about the effectiveness of our approach.

6.2 The Greek Bailout Referendum

The period of the Greek economic crisis before the bailout referendum (2009-2015) was characterized by extreme political turbulence, when Greece faced six straight years of economic recession and five consecutive years under two bailout

²As per Twitter Streaming API limitations: <https://developer.twitter.com/en/docs/basics/rate-limiting>

programs [246]. Greek governments agreed to implement austerity measures, in order to secure loans and avoid bankruptcy – a fact that caused massive unrest and demonstrations. During the same period, political parties regardless of their side on the left-right political spectrum were divided into *pro-austerity* and *anti-austerity*, while the traditional two-party system conceived a big blow [28, 236, 212].

The Greek bailout referendum was announced on June, 27th 2015 and was held eight days later. The Greek citizens were asked to respond as to whether they agree or not (YES/NO) with the new bailout deal proposed by the Troika³ to the Greek Government in order to extend its credit line. The final result was 61.3%-38.7% in favor of the NO vote. For more details on the Greek crisis, refer to Tsebelis [246].

6.3 Task Description

Our aim is to classify a Twitter user either as a YES or a NO voter in the Greek Bailout referendum over the eight-day period starting right before its announcement (26/6, day 0) and ending on the last day before it took place (4/7, day 8). Due to the very short duration of the pre-electoral race, we consider the user’s stance (YES or NO) as a static variable we aim at predicting; however we update the input to our models (and re-assess their performance) with new user-generated data that was shared on a daily basis, thus mimicking a real-world and real-time political monitoring setting.

We assume a training set of users:

$$D_t = \{(x_t^{(1)}, y^{(1)}), \dots, (x_t^{(n)}, y^{(n)})\},$$

where $x_t^{(i)}$ is a representation of user i up to time step $t \in [0, \dots, 8]$ and $y^{(i)} \in \{\text{YES}, \text{NO}\}$. Given D_t , we want to learn a function f_t that maps a user j to her

³A decision group formed by the European Commission, the European Central Bank and the International Monetary Fund to deal with the Greek economic crisis.

or his stance at time t :

$$\hat{y}^{(j)} = f_t(x_t^{(j)}).$$

Then, we update our model with new information shared by the users in our training set up to $t+1$, to predict the test users voting intention at $t+1$. For example, when $t = 2$, we predict the stance of users using the information in $[0, 1, 2]$ where 0 represents the information available up to the day before the announcement; afterwards, for $t = 3$, we predict the stance of users using information in $[0, 1, 2, 3]$, and so forth. Therefore, we mimic a real-time setup, where we nowcast user voting intention, starting from the moment before the announcement of the referendum, until the day of the referendum. Sections 6.4 and 6.5 present how we develop the training dataset D_t and the function f_t respectively.

6.4 Data

Using the Twitter Streaming API during the period 18/6–16/7, we collected 14.62M tweets in Greek (from 304K users) containing at least one of 283 common Greek stopwords, starting eight days before the announcement of the referendum and stopping 11 days after the referendum date (see Figure 6.1). This provides us with a rare opportunity to study the interaction patterns among the users in a rather complete and unbiased setting, as opposed to the vast majority of past works, which track event-related keywords only. For example, Antonakaki et al. [9] collected 0.3M tweets using popular referendum-related hashtags during 25/06–05/07 – we have collected 6.4M tweets during the same period. In the rest of this section, we provide details on how we processed the data in order to generate our training set in a semi-supervised way (6.4.1) and how we annotated the users that were used as our test set in our experiments (6.4.2).

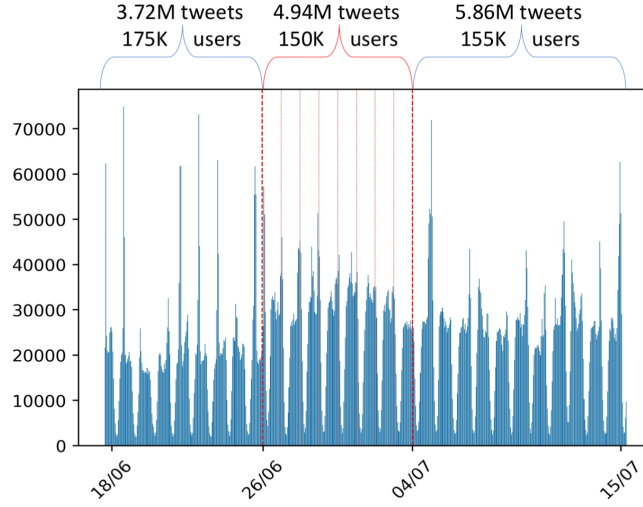


Figure 6.1: Number of tweets in Greek per hour. The period highlighted in red indicates the nine evaluation time points, starting before the announcement of the referendum and ending in the day before the date of the referendum.

6.4.1 Training Set

Manually creating a training set would have required annotating users based on their voting preference on an issue that they had not been aware of prior to the referendum announcement. However, the same does not hold for certain accounts (e.g., major political parties) whose stance on austerity had been known a-priori given their manifestos and previous similar votes in parliament [212]. Such accounts can be used as seeds to form a semi-supervised task, under the hypothesis that users who are re-tweeting a political party more often than others, are likely to follow its stance in the referendum, once this is announced. Hence, we compile a set of 267 seed accounts (148 YES, 119 NO) focusing on the pre-announcement period including: (1) *political parties*; (2) *members of parliament (MPs)*; and (3) *political party members*.

- **Political Parties:** We add as seeds the Twitter accounts of nine major and minor parties⁴ with a known stance on austerity before the referendum (5 YES, 4 NO, see Table 6.1). We assume that the pro-austerity parties will

⁴We excluded KKE (Greek Communist Party) since an active official Twitter account did not exist at the time.

Table 6.1: Political position, austerity, referendum stance and national election result (January 2015) of the political parties that are used as seeds in our modelling.

Party	Position	Austerity	Referendum	Jan 15 (%)
SYRIZA	Left	anti	NO	36.34
New Democracy	Centre-Right	pro	YES	27.81
Golden Dawn	Far-right	anti	NO	6.28
The River	Centre	pro	YES	6.05
Independent Greeks	Right	anti	NO	4.75
PASOK	Centre-left	pro	YES	4.68
KIDISO	Centre-left	pro	YES	2.47
ANTARSYA	Far-left	anti	NO	0.64
Dimiourgia Xana	Centre-Right	pro	YES	-

back the bailout proposal (YES), while the anti-austerity parties will reject it (NO). The pro-/anti- austerity stance of the parties was known before the referendum, since the pro-austerity parties had already backed previous bailout programs in parliament or had a clear favorable stance towards them, whereas the opposite holds for the anti-austerity parties [212].

- **MPs** The accounts of the (300) MPs of these parties were manually extracted and added as seeds. 153 such accounts were identified (82 YES, 71 NO) labelled according to the austerity stance of their affiliated party.
- **Political Party Members** We finally compiled a set of politically-related keywords to look up in Twitter user account names and descriptions (names and abbreviations of the nine parties and keywords such as “candidate”). We identified 257 accounts (133 YES, 124 NO), which were manually inspected by human experts to filter out irrelevant ones (e.g., the word “River” might not refer to the political party) and kept only those that had at least one tweet during the period preceding the announcement of the referendum (44 NO, 61 YES).

To expand the set of seed accounts, we calculate for every user u in our dataset during the pre-announcement period his/her score as:

$$score(u) = PMI(u, YES) - PMI(u, NO),$$

Table 6.2: Number of users (u) and tweets (t) used in our experiments per evaluation day.

day	0	1	2	3	4	5	6	7	8
date	26/06	27/06	28/06	29/06	30/06	01/07	02/07	03/07	04/07
train_u	2121	2121	2121	2121	2121	2121	2121	2121	2121
train_t	307K	395K	468K	543K	609K	685K	752K	814K	867K
test_u	1804	1985	2045	2115	2146	2174	2184	2194	2197
test_t	293K	358K	414K	477K	533K	599K	658K	718K	768K

where $PMI(u, lbl)$ is the pointwise mutual information between a certain user and the respective seeding class (YES/NO). A high (low) score implies that the user is endorsing often YES-related (NO-related) accounts, thus he/she is more likely to follow their stance after the referendum is announced. This approach has been successfully applied to other related natural language processing tasks, such as building sentiment analysis lexical resources using a pre-defined list of seed words [160]. Assigning class labels to the users based on their scores, we set up a threshold:

$$tr = n(\max(|scores|)), n \in [0, 1].$$

We assign the label YES to a user u iff $score(u) > tr$ or NO iff $score(u) < -tr$. Setting $n = 0$, would imply that we are assigning the label YES if the user has re-tweeted more YES-supporting accounts (and inversely), which might result into a low quality training set, whereas higher values for n would imply a smaller (but of higher quality) training set. During development, we empirically set $n = 0.5$ to keep users who are fairly closer to one class than the other. From the final set of 5,430 users that have re-tweeted any seed account, 2,121 were kept (along with the seed accounts) as our training set (965 YES, 1,156 NO).

6.4.2 Test Set

For evaluation purposes, we generate a test set of active users that are likely to participate in political conversations on Twitter. First, we identify all users having tweeted at least 10 times after the referendum announcement (86,000 users).

From the 500 most popular hashtags in their tweets, we selected those that were clearly related to the referendum (189) which were then manually annotated by experts with respect to potentially conveying the user’s voting intention (e.g., “yesgreece”, “no” as opposed to neutral ones, such as “referendum”). Finally, we selected a random sample of 2,700 users (out of 22K) that had used more than three such hashtags, to be manually annotated – without considering any user from the training set. This is standard practice in related work [77, 228] and enables us to evaluate our models on a high quality test set, as opposed to previous related work which rely on keyword-matching approaches to generate their test set [72, 263].

Two experts (Greek native speakers) annotated each of the users in the test set, using the tweets after the referendum announcement. Each annotator was allowed to label an account as YES, NO, or N/A, if uncertain. There was an agreement on 2,365 users (Cohen’s $\kappa = .75$) that is substantially higher if the N/A labels are not considered ($\kappa = .98$), revealing high quality in the annotations, i.e., in the upper part of the ‘substantial’ agreement band [11]. We discarded all accounts labelled as N/A by an annotator and used the remaining accounts where the annotators agreed for the final test set, resulting to 2,197 users – similar test set sizes are used in related tasks [62]. The resulting user distribution (NO 77%, YES 23%) is more imbalanced compared to the actual result of the referendum, due to the demographic bias on Twitter [156]. To mimic a real-time scenario, we refrained from balancing our train/test sets, since it would have been rather impossible to know the voting intention distribution of Twitter users a-priori. Overall, we use 18.9% (1.64M/8.66M) of the tweets written in Greek during that period in our experiments (see Table 6.2).

6.5 Models

6.5.1 Convolution Kernels

Convolution kernels are composed of sub-kernels operating on the item-level to build an overall kernel for the object-level [92, 46] and can be used with any kernel based model such as Support Vector Machines (SVMs) [108]. Such kernels have been applied in various NLP tasks [46, 118, 249, 141]. Here we build upon the approach of Lukasik and Cohn [141] by combining convolution kernels operating on available (1) *text*; and (2) *network information*.

Let a, b denote two objects (e.g., social network users), represented by two $M \times N$ matrices Z_a and Z_b respectively, where M denotes the number of items representing the object and N the dimensionality of an item vector. For example, an item can be a user's tweet or network information. A kernel K between the two objects (users) a and b over Z_a and Z_b is defined as:

$$K_z(a, b) = \frac{1}{|Z_a||Z_b|} \sum_{i,j} k_z(z_a^i, z_b^j), \quad (6.1)$$

where k_z is any standard kernel function such as a linear or a radial basis function (RBF). One can also normalise K_z by dividing its entries $K_z(i, j)$ by $\sqrt{K_z(i, i)K_z(j, j)}$.

The resulting kernel has the ability to capture the similarities across objects on a per-item basis. However, unless restricted to operate on consecutive items (time-wise), it ignores their temporal aspect. Given a set of associated timestamps $T_o = \{t_o^1, \dots, t_o^N\}$ for the items of each object o , Lukasik and Cohn [141] proposed to combine the temporal and the item aspects as:

$$K_{zt}(a, b) = \frac{1}{|Z_a||Z_b|} \sum_{i,j} k_z(z_a^i, z_b^j) k_t(t_a^i, t_b^j), \quad (6.2)$$

where k_t is any valid kernel function operating on the timestamps of the items. Here, K_{zt} is a matrix capturing the similarities across users by leveraging both the information between pairs of items and their temporal interaction.

6.5.2 Text Kernels

Let a, b denote two users in a social network, posting messages $W_a = \{w_a^1, \dots, w_a^N\}$ and $W_b = \{w_b^1, \dots, w_b^M\}$ with associated timestamps $T_a = \{t_a^1, \dots, t_a^N\}$ and $T_b = \{t_b^1, \dots, t_b^M\}$ respectively. We assume that a message w_i^j of user i at time j is represented by the mean k -dimensional embedding [153] of its constituent terms. This way, we can obtain *text convolution kernels*, K_w and K_{wt} by simply replacing Z and z with W and w respectively in Equations 6.1 and 6.2. Following [141], we opted for a linear kernel operating on text and an RBF on time.

6.5.3 Network Kernels

Let assume a set of directed weighted graphs:

$$G = \{G_1(N_1, E_1), \dots, G_t(N_t, E_t)\}$$

where $G_i(N_i, E_i)$ represents the retweeting activity graph of the N_i users at a time point $i \in T = \{1, \dots, t\}$. Let $L_a \in \mathbf{R}^{N,k}$, $L_b \in \mathbf{R}^{M,k}$ denote the resulting matrices of a k -dimensional, network-based user representation for two users a and b across time. Contrary to the textual vector representation w_i^j that is defined over a fixed space given a pre-defined vocabulary, user network vector representations (e.g., graph embeddings [232]), are computed at each time step on a different network structure. Thus, a standard similarity score between two user representations at timepoints t and $t+1$ cannot be used, since the network vector spaces are different. To accommodate this, at each time point t we calculate the median L_{YES}^t and L_{NO}^t vectors for each class of our training examples and update the respective user vectors as:

$$L_u^{*t} = d(L_{\text{YES}}^t, L_u^t) - d(L_{\text{NO}}^t, L_u^t),$$

using some distance metric d (for simplicity, we opted for the linear distance). If a user has not retweeted, his/her original network representation l_u^t is calculated as the average across all user representations at t . Finally, the *network convolution kernels*, K_n and K_{nt} are computed using Equations 6.1 and 6.2 respectively by simply replacing Z with L^* and z with l^* . Similarly to text kernels, we use a linear kernel k_n for the network and an RBF kernel k_t for time.

6.5.4 Kernel Summation

We can combine the text and network convolution kernels by summing them up: $K_{sum} = K_w + K_{wt} + K_n + K_{nt}$. This implies a simplistic assumption that the contribution of the different information sources with respect to our target is equal. While this might hold for a small number of carefully designed kernels, it lacks the ability to generalise over multiple kernels of potentially noisy representations.

6.5.5 SVMs with Convolution Kernels

Convolution kernels can be used with any kernel based model. Here, we use them with SVMs. First, a SVM_s operates on a single information source $s = \{w, n\}$, i.e., SVM_w for text and SVM_n for network. Second, a SVM_{st} takes temporal information into account combined with text (SVM_{wt}) and network (SVM_{nt}) information respectively. Finally, we combine the text and the network information using a linear kernel summation (K_{sum}) of their respective kernels (SVM_{sum}).

6.5.6 Multiple Convolution Kernel Learning (MCKL)

Multiple kernel learning methods learn a weight for each kernel instead of assigning equal importance to all of them allowing more flexibility. Such approaches have been extensively used in tasks where different data modalities exist [104, 242, 196]. We build upon the approach presented in [227] to build a

model based on labelled instances $x_i \in I$, by combining the different convolution kernels K_s with some weight $w_s > 0$ s.t. $\sum_s w_s = 1$ and apply:

$$f(x) = \text{sign}\left(\sum_{i \in I} \alpha_i \sum_s w_s K_s(x, x_i) + b\right)$$

As presented in Chapter 2, the parameters α_i , the bias term b and the kernel weights are estimated by minimising the expression:

$$\min \quad \gamma - \sum_{i \in I} \alpha_i \tag{6.3}$$

$$\text{w.r.t.} \quad \gamma \in R, \alpha \in R_+^{|I|}$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq C \quad \forall i, \quad \sum_{i \in I} \alpha_i y_i = 0$$

$$\frac{1}{2} \sum_{i \in I} \sum_{j \in I} \alpha_i \alpha_j y_i y_j K_s(\mathbf{x}_i, \mathbf{x}_j) \leq \gamma \quad \forall s \tag{6.4}$$

$$\tag{6.5}$$

This way, the four convolution kernels are calculated individually and subsequently combined in a weighted scheme accounting for their contribution in the prediction task. This allows us to combine external and asynchronous information (e.g., news articles), while adding other kernels capturing different aspects of the users (e.g., images) is straight-forward.

6.6 Experimental Setup

6.6.1 Features

We extract features derived from the tweets (“TEXT”) and the re-tweeting activity (“NETWORK”) of the users in our training and test set. In our experiments, we use each of these two sources as input to various models individually, as well as in concatenation (“BOTH”).

Textual Information (TEXT) We obtain word embeddings by training `word2vec` [153] on a collection of 14.7 non-retweeted tweets (i.e., we excluded retweets, so that we do not have repetitions of the exact same content), collected in the exact same way as our dataset, over a separate period of 2 months. We performed standard pre-processing steps including lowercasing, tokenising, removal of non-alphabetic characters, replacement of URLs, mentions and all upper-case words with identifiers. We used the CBOW architecture, opting for a 5-token window around the target word, discarding all words appearing less than 5 times and using negative sampling with 5 “noisy” examples. After training, each word is represented as a 50-dimensional vector. Each tweet in our training and test set is represented by averaging each dimension of its constituent words.

Network Information (NETWORK) We trained LINE [232] embeddings at different timesteps, by training on the graphs $\{G_1(N_1, E_1), \dots, G_T(N_T, E_T)\}$, where N_i is the set of users and E_i is the (directed, weighted) set of retweets amongst N_i up to time i . We choose the “retweet” rather than the “user mention” network, since retweets are more likely to be endorsements⁵. LINE was preferred over alternative models [190, 210] due to its ability to model directed weighted graphs. We construct the network G_t every 12 hours based on the retweets among all users up to time t , and LINE is trained on G_t to create 50-dimensional user representations. We used the second-order proximity, since it performed better than the first-order in early experimentation. We also refrained from concatenating them to keep the dimensionality relatively low. An alternative approach would have been to construct the network in a sliding window approach; however, finding the optimal value for its width can be a crucial context-dependent task, which opposes our goal of generalisation ability. The charts in Figure 6.2 provide some basic descriptive attributes of the constructed

⁵The “following” network cannot be constructed based on the JSON objects returned by Twitter Streaming API; to achieve this requires a very large number of API calls and cannot be constructed accurately in a realistic scenario.

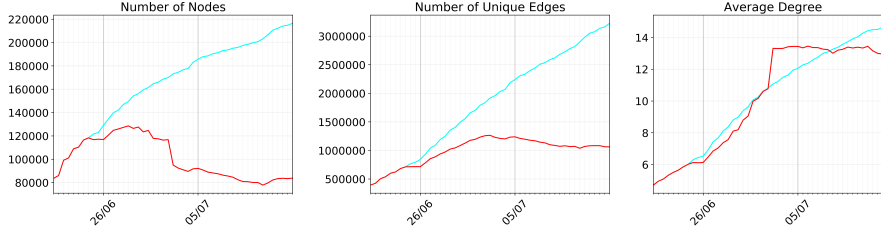


Figure 6.2: Number of nodes N (users), re-tweet edges E and average degree ($|E|/|N|$) of the re-tweeting network over time, in a cumulative fashion (blue) and in a sliding window approach (red) of network construction (i.e., based on the re-tweeting activity of the past seven days).

re-tweet graphs over time, when those are generated in a cumulative (as in our modelling) as well as in a sliding window fashion (i.e., by considering only the previous seven days to construct the network at each time point). The increase in the graph connectivity over the period between the announcement of the referendum and until the date that it took place is depicted by the increase in the number of edges and the average degree over this time period. In our modelling, we accompany the textual features with such network-based and time-sensitive information, albeit in a user-specific manner, in order to study their effectiveness on our task.

6.6.2 Models

Convolution Kernel Models Our MCKL and our SVM models are fed with the convolution kernels operating on the tweet-level (for TEXT) and each NETWORK representation (derived every 12 hours), based on the tweets and re-tweeting activity respectively of the users up to the current evaluation time point.

Baselines We compare our proposed methods against competitive baselines that are commonly used in social media mining tasks trained on feature aggregates [142, 263]. We obtain a TEXT representation of a user at each time step t by averaging embedding values across all his/her tweets until t . Similarly, a

user NETWORK representation is computed from the retweeting graph up until t . Finally, we train a regularised Logistic Regression (**LR**) with L_2 regularisation [133], a feed-forward neural network (**FF**) [98], a Random Forest (**RF**) [32] and a **SVM**.

Model Parameters Parameter selection of our models and the baselines is performed using a 5-fold cross-validation on the training set. We experiment with different regularisation strength ($10^{-3}, 10^{-2}, \dots, 10^3$) for LR, different number of trees (50, 100, ..., 500) for RF, and different kernels (linear, RBF) and parameters C and γ ($10^{-3}, 10^{-2}, \dots, 10^3$) for SVMs. For FF, we stack dense layers, each followed by a ReLU activation and a 20% dropout layer, and a final layer with a sigmoid activation function. We train our network using the Adam optimiser [120] and experiment with different number of hidden layers (1, 2), units per layer (10, 25, 50, 75, 100, 150, 200), batch size (10, 25, 50, 75, 100) and number of epochs (10, 25, 50, 100). For MCKL, we experiment with the same C values as in SVM and apply an L_2 regulariser.

6.6.3 Evaluation

We train and test our models based on the data collected on a daily basis (every midnight), starting from the day before the announcement of the referendum (day 0) until the day before its due date (day 8), aiming to classify the test users' (static) voting intention in such a temporal manner. This way, we mimic a real-time setting and gain better evaluation insights. To evaluate our models, we compute the macro-average F-score, which forms a more challenging metric compared to micro-averaging, given the imbalanced distribution of our test set. Parameter selection is performed on every evaluation day using a 5-fold cross-validation on the training set. At each evaluation time point t , we only classify the users that have tweeted at least once up to t . This results into a different number of test instances per day (see Table 6.2). However, we did not observe any major differences in our evaluation by excluding newly added users. In cases

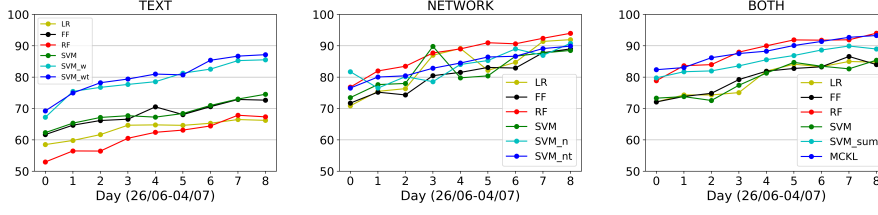


Figure 6.3: Macro-average F-score across all evaluation days using **TEXT**, **NETWORK** and **BOTH** user representations.

where a certain user had tweeted up to an evaluation time point t but he/she had not re-tweeted any other user, we model his/her **NETWORK** representation as the average **NETWORK** representation across all users at this point in time.

6.7 Results

In the current section we present the results of our MCKL on the voting intention task, compared to the feature aggregate baselines presented in 6.6. We then demonstrate the robustness of our approach, when dealing with noisy modalities.

6.7.1 Nowcasting Voting Intention

Figure 6.3 presents the macro-average F-scores obtained by the methods compared in all days from the announcement to the day of the referendum. As expected, the closer the evaluation is to the referendum date, the more accurate the models since more information becomes available for each user. Table 6.3 shows the average (across-all-days) F-score by each model.

Temporal convolution kernels using **TEXT** (SVM_{wt}) significantly outperform the best text-based baseline ($p = .001$, Kruskal-Wallis test against SVM), with an average of 11.8% and 17.2% absolute and relative improvement respectively. This demonstrates the model’s ability on capturing the similarities between different users on a per-tweet basis compared to simpler models using tweet aggregates. Also, SVM_w and SVM_{wt} implicitly capture similarities in the retweeting activity of the users. This is important, since network information

Table 6.3: Average F-score and standard deviation across all evaluation days using **TEXT**, **NETWORK** and **BOTH** user representations. SVM_s and SVM_{st} denote the SVM with convolution kernels (SVM_w , SVM_n) and (SVM_{wt} , SVM_{nt}), respectively.

	TEXT	NETWORK	BOTH
LR	63.55 \pm 2.86	83.21 \pm 7.55	79.43 \pm 5.34
FF	68.19 \pm 3.78	80.66 \pm 5.93	79.83 \pm 5.11
RF	61.27 \pm 5.14	87.43 \pm 5.60	88.22 \pm 5.03
SVM	68.51 \pm 3.80	82.43 \pm 5.83	79.39 \pm 5.18
SVM_s	78.91 \pm 5.68	83.65 \pm 4.82	–
SVM_{st}	80.30 \pm 5.81	84.03 \pm 4.47	–
SVM_{sum}	–	–	85.22 \pm 3.64
MCKL	–	–	88.31 \pm 3.95

might not be easily accessible (e.g., due to API limitations) while it is expensive to compute at each timestep. Hence, one can use SVM_{wt} to model user written content and partially capture network information (i.e. retweets are included in **TEXT** representation).

Classification accuracy consistently improves when using the **NETWORK** representation (i.e., graph embeddings). RF achieves 94% F-score on the day before the referendum, whereas the worst-performing baseline (FF) still achieves 80.66% F-score on average. SVM_{nt} provides a small boost (1.6% on average) compared to the vanilla SVM, which uses only the user representations derived at the current time point. This implies that the current network structure is indicative of users’ voting intention, probably because the referendum was the dominant topic of discussion at the time, e.g., most of the retweeting activity was relevant the referendum (see Section 6.8). Even right before the announcement of the referendum, our baselines achieve 73% F-score on average. The corresponding average of our two convolution kernel approaches is even higher (79.1%). We should note that our test set includes users with high and regular activity in Twitter that might be easier to model, however it is rather impossible to annotate the voting intention of users with no or minimal activity.

When combining the user text and network representation (**BOTH**), the baselines fail to improve over using only **NETWORK**. In contrast, our MCKL improves by 4.28% over the best performing single convolution kernel model

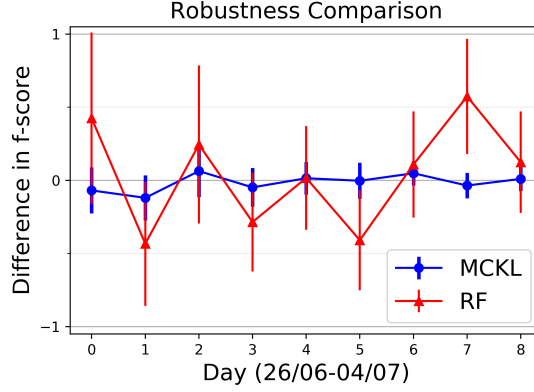


Figure 6.4: Change in performance (mean/standard deviation) compared to the results in Figure 6.3, after 100 experiments with added noisy features.

(SVM_{nt}). This demonstrates that MCKL can effectively combine information from both representations by weighting their importance, and further improve the accuracy of the best performing single representation model. Overall, MCKL significantly outperforms the best performing text-based baseline by approximately 20% in F-score ($p < .001$, Kruskal-Wallis test).

6.7.2 Robustness Analysis

Due to the semi-supervised nature of our task, it is impossible to judge whether the small difference between MCKL and RF stems from a better designed model. Furthermore, it is difficult to assess MCKL’s effectiveness with respect to its ability to generalise over multiple and potentially noisy feature sources.

To assess the robustness of the best performing models (MCKL, RF) operating on **BOTH** information sources, here we perform experiments by adding random noise in their input. We assume that there is a noisy source generating an extra K -dimensional representation X for every user that we add as extra input to the models. We set $K = 25$, so that (a) we account for a smaller noisy input compared to our features (25 vs 50) and (b) 1/5 of our kernels in MCKL and 25/125 input features in RF are noisy. We perform 100 runs, each time drawing random noise $X \sim N(0, 1)$.

Our results indicate that RF is more sensitive to the noisy input compared to MCKL (see Figure 6.4). On average, RF achieves a small boost (0.04%) in performance with the added noise. That together with the higher standard deviation reveal the vulnerability of RF to potentially corruption and stochasticity introduced in the input. On the contrary, MCKL is consistently robust, achieving only a tiny reduction in performance on average across all days (0.02%) while the respective average standard deviation is lower than the one achieved by RF (0.12 vs 0.41). This robustness is highly desirable in cases of such sudden political events and it also indicates that we can add kernels capturing different properties of our task (e.g., user-related information, images, etc.), without having to decide a-priori which of them are indeed predictive of the user’s voting intention.

6.8 Qualitative Analysis

Besides evaluating MCKL, we are also interested in providing insights into the temporal variation observed in the users’ shared content and the network structure during this major political crisis. In the current section we provide details on both of these aspects.

6.8.1 Language

We are interested in investigating which are the political-related entities that voters from both sides most likely mention. We expect that this will shed light on the main focus of discussion in the political debates between the YES/NO voters that occurred after the announcement of the referendum. For this, two experts manually compiled two lists of n-grams containing different ways of referring⁶ to the (a) the six major political parties and (b) their leaders (see Table 6.1). We represent every YES/NO user in the test set as aggregated `tf-idf` values of the ngrams (1-3) appearing in his/her concatenated tweets; then, we

⁶Note that Greek is a fully inflected language. We opted not to apply stemming because inflected word forms carry meaningful information.

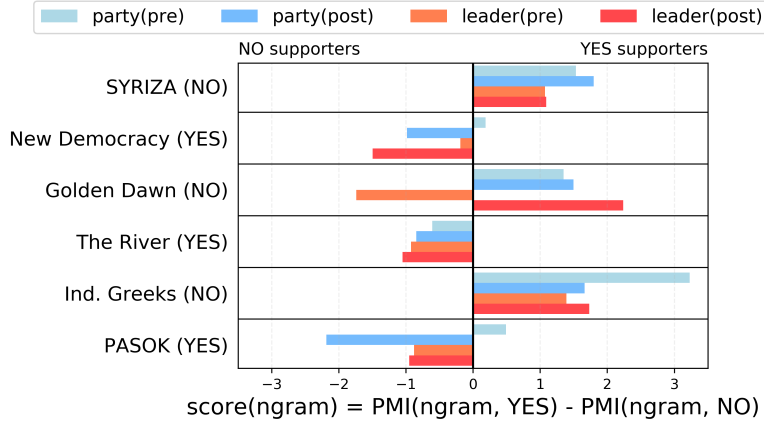


Figure 6.5: Scores of n-grams related to the political parties/leaders, pre (18/06-26/06) and post (27/06-05/07) the referendum announcement. Scores < 0 (> 0) indicate that n-grams appear mostly in tweets of NO (YES) voters.

Table 6.4: Most similar words to YES and NO (translated to English), when training word2vec on different time periods.

	Before the announcement (18/06-26/06)	After the announcement (27/06-05/07)
YES	no, ok, nah, alright, sure, usmnt, hahaha, alrighty, but, so	no, abstain, referendum, KKE, question, invalid, euro, clearly, clash, nai
NO	yes, only, sure, so (slang), disagree, mainly, especially, obviously, so (abbrv), agree	yes, abstain, KKE, referendum, clash, question, people, invalid, vote, clearly

compute an n-gram's n score as $PMI(n, \text{YES}) - PMI(n, \text{NO})$. A positive score implies that it is highly associated with users who support the YES vote, and vice versa.

Figure 6.5 shows that the parties and leaders that supported one side, mostly appear in tweets of users supporting the opposite side. This is more evident when we consider tweets shared by the users *after* the announcement of the referendum. Examining the content of highly-retweeted tweets, revealed sarcasm and hostility for the opposite side in the majority of them (see Table 6.5). Hostility is a frequent phenomenon in public debates [109] and our findings corroborate previous work showing that the political discourse on Twitter is polarised [47, 77].

Table 6.5: Examples of highly re-tweeted tweets after the announcement of the referendum.

Tweet	#RT
<i>They say that there is a long queue of people in ATMs but they show only 6 people waiting; this is not a queue, this is <u>PASOK</u>.</i>	686
<i>See this photo (attached) so that you know who is in charge of the far-right <u>New Democracy</u> of <u>Samaras</u>...</i>	520
<i>Looking for any angry tweets by <u>SYRIZA</u> fans concerning <u>Kassidiaris's</u> (Golden Dawn MP) release from prison. Have you seen any?</i>	246
<i>I want to write something funny regarding the statements made by <u>Kammenos</u> (Ind. Greeks leader), but I cannot find something funnier than the statements made by <u>Kammenos</u>.</i>	178
<i>Now you can see why the European leaders wanted <u>The River</u> to be in the government coalition.</i>	120

Finally, we examine the temporal variation of language over the same two periods. Table 6.4 shows the most similar words (translated to English) to the *yes* and *no* words, measured by cosine similarity, when training `word2vec` using the tweets of each time period. The difference of the cosine similarities $\cos_{post} - \cos_{pre}$ between the *yes/no* vectors and each of their corresponding most similar words over these two periods is shown in Figure 6.6. After the announcement, the context of the two words shifts towards the political domain. That might explain why text aggregates become noisy, as shown in our results. Convolution kernels are able to filter-out this noise since they operate on the tweet level by also taking the time into account. We plan to study the semantic variation in language [60] in a more fine-grained way in future work.

6.8.2 Network

We explore the differences in retweeting behaviour of users over the same periods ((a) before the announcement of the referendum and (b) after and until the day of the referendum), by training two different LINE embedding models using tweets from the each period respectively. Figure 6.7 shows the plots of the first two dimensions of the graph embeddings before and after the an-

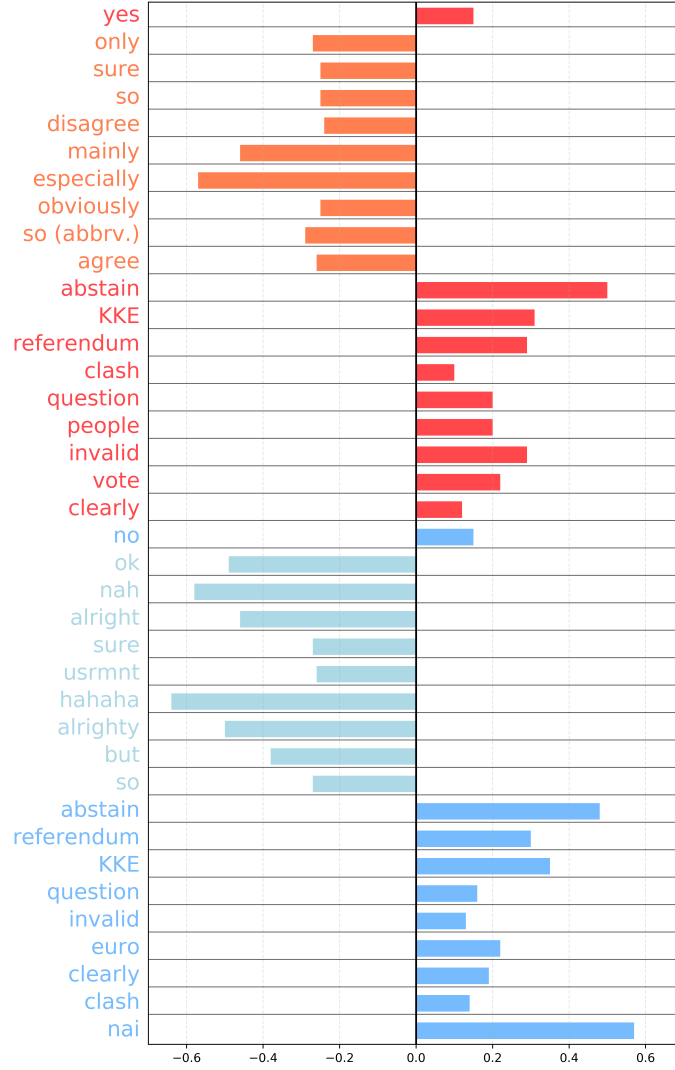


Figure 6.6: Difference in cosine similarity ($\cos_{post}(w_{no/yes}, w) - \cos_{pre}(w_{no/yes}, w)$) between the *no/yes* (red/blue) word vectors $w_{no/yes}$ and each of their most similar words in the two periods.

nouncement using principal component analysis. The results unveil the effects of the referendum announcement and provide insights on the effectiveness of NETWORK information for predicting vote intention, as demonstrated in our results. Before, YES and NO users appear to have similar retweeting behaviour, which changes after the announcement. This finding illustrates the political homophily of the social network [45] and highlights the extremely polarised

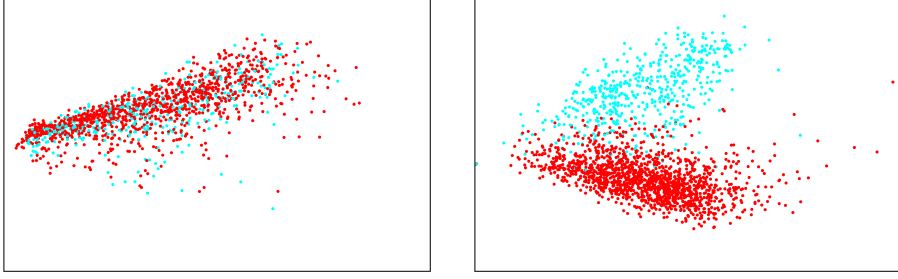


Figure 6.7: Network representations of YES/NO (blue/red) users, before (left) and after (right) the referendum announcement.

pre-election period [246].

Next, we question whether the distance between the two classes of users through time changes according to time points at which real-world events occur. To answer this, we compute the network embeddings of the train and test users every 12 hours, as in our experiments, and represent every class (YES/NO) at a certain time point t by the average representations (avg_Y^t, avg_N^t) of the corresponding users in the training set at t . Then, for every user u in the test set, we use the cosine similarity cos to calculate:

$$network_score_u^t = cos(u^t, avg_Y^t) - cos(u^t, avg_N^t)$$

Finally, we calculate the average score of the YES and the NO users in the test set $(network_Y^t, network_N^t)$ at every time point t and normalise the corresponding time series s.t. $network_Y(0) = network_N(0) = 0$. We also employ an alternative approach, by generating the network embeddings on a seven-day sliding-window fashion and following the same process. The results are shown in Figure 6.8. In both cases, the YES/NO users start to deviate from each other right after the announcement of the referendum, with an upward/downward YES/NO trend until the day of the referendum. This is effectively captured in our modelling and might explain the reason for the high accuracy achieved even by our baseline models, which are trained using the network representation of the users in the last day only. However, the YES/NO users start to again ap-

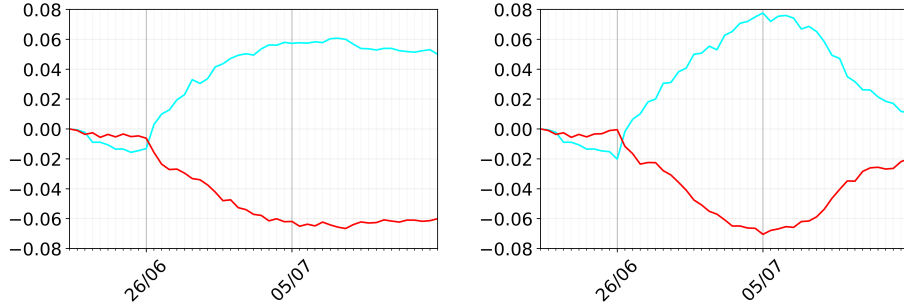


Figure 6.8: Normalised difference of similarity of YES/NO (blue/red) users in our modelling (left) and in a sliding window approach (right).

proach each other only in the sliding window approach after the referendum day, since in our modelling the representations are built based on re-tweets aggregates over the whole period. It is apparent that such a temporal modelling of the network structure, as depicted in the right of Figure 6.8, would yield more accurate predictions in longer lasting electoral cases, since the deviation of the two classes of users would be picked up more effectively by calculating their similarities with respect to their network structure across time. While this does not seem to have affected our performance, exploring the temporal structure of the network formations through time is of vital importance for longer lasting electoral cases.

6.9 Summary and Conclusion

In this chapter, we presented a distant-supervised multiple convolution kernel approach, leveraging temporally sensitive language and network information to nowcast the voting stance of Twitter users during the 2015 Greek bailout referendum. Importantly, our approach can easily be adjusted to other cases of political debates, due to its language-independent nature. Furthermore, the modelling of text and network structure in a temporal fashion as presented in this chapter has the potential to yield more robust and accurate predictions of a user’s stance in longer lasting cases of debates, since it has the ability to capture

similarities across different users over time.

This was the first step towards our *micro-level* goals, effectively addressing RQ2. However, in this chapter we dealt with a time-invariant target (i.e., voting intention) per user, while we only explored one source of information (i.e., social media). As more modalities (e.g., smart phones, wearable sensors, etc.) become available in the real-world, finding effective ways to make use of such heterogeneous information sources is becoming of high research interest. We are investigating such cases in the micro-level, in the chapters that follow up next.

CHAPTER 7

Micro-level Modelling Using Heterogeneous Data Sources

RQ3. *Can we use asynchronous and heterogeneous data streams from a specific group of users in order to nowcast their temporally sensitive real-world indices on the micro-level?*

In this chapter, we expand our methodology from the previous chapter, to account for (a) a time-varying target and (b) heterogeneous sources of information. Due to the lack of such data – for both (a) and (b) – in our previously studied domain (i.e., the political domain), we switch our application to the mental health domain. In particular, we work on a longitudinal dataset derived by a cohort of 19 students, aiming to predict their mental health indices on a longitudinal basis. From a textual perspective, this dataset contains posts and private messages from the social media accounts of the subjects, as well as their private SMS messages. These are accompanied by sensor-based information and logs derived from their smartphones. Such asynchronous and heterogeneous information is used to predict the subjects’ mental health indices, which are derived in the basis of two well-established psychological scales. We propose a MKL regression model, modelling every modality via a different kernel, and compare our approach against various baselines. Finally, we provide evidence on the types of features that are most predictive of a subject’s mental health, as derived from our MKL model, and the effectiveness of MKL to assess mental health by making better use of heterogeneous and complementary modalities, as opposed to baseline approaches¹.

¹The current chapter is based on [242].

7.1 Introduction

The World Health Organisation describes mental health as “the foundation for well-being and effective functioning for an individual and for a community” and highlights the importance of selecting suitable indicators of mental health [94]. Poor mental health is highly correlated with low motivation, lack of satisfaction, low productivity and a negative economic impact [175]. One can distinguish between macro-level indicators, which are meant to provide a picture of generic well-being across a large population, usually at national scale, and individual indicators of mental health. Most of the macro measures typically use statistics from census, administrative and economic sources to measure the social and economic macro-environment as important determinants of mental health (e.g., Human Development Index, Gender Development Index, Human Poverty Indices [174]). With the advent of widely available social media data, there have also been efforts to automatically obtain macro indicators of well-being and happiness, primarily through the analysis of geolocated Twitter posts [65, 128, 125]. These pieces of work seek to identify occurrence patterns for words with pre-defined affect scores at different levels of temporal granularity. Such approaches, with more sophisticated components for emotion recognition in social media content, can be alternatives to public surveys for mood and happiness indicators.

At the other end of the spectrum we have individual indicators of mental health. These include measures of positive mental health, such as coherence & meaning in life, self-esteem etc. as well as indicators of mental distress, such as negativity, anxiety, depression [94]. These measures can be used by experts or individuals for diagnostic and management purposes, but also in aggregation, for large scale surveys. However, the reliance on self-reporting required to obtain these measures is time consuming and expensive and can only produce sparse data on small populations. Moreover, self-reporting is likely to introduce bias into results. Recent work [204, 130, 36, 185] shows the potential of experience sampling using mobile devices for behavioural studies and clinical

care, especially relating to mental health. A variety of longitudinal sensor data from a smart phone as well as location information, obtained passively from the user’s phone, can be calibrated against the user’s responses to behaviour or emotion related questions. The latter are usually harvested through regular prompts for input provided by a smart phone application.

Here we combine heterogeneous and asynchronous textual as well as non-linguistic data to train predictors of well-being scores that will circumvent the need for user input.

Our contributions include:

- **A novel and unique dataset of heterogeneous sources** consisting of textual data from social media posts (Twitter, Facebook), SMS messages ($> 100,000$), 2436 mood forms as well as asynchronous mobile phone use data including location, Wi-Fi connection, mobile phone use and sensor data (42 GB).
- **Methodology for handling heterogeneous, incomplete and asynchronous data for longitudinal predictions.** We consider a number of baselines and appropriate normalisations as well as an approach based on multi-kernel learning, which aims to maximise the joint predictive power of each data source, and show very promising results.
- **Calibration of well-being predictors based on well established affect and well-being scales**, namely the Warwick-Edinburgh Mental Well-Being Scale (WEMWBS) [235] and the Positive and Negative Affect Scale (PANAS) [257, 54].

While studies on macro-indicators have exploited simple textual features, we are not aware of another study which has worked on such an heterogeneous dataset for the automatic prediction of individual well-being scores, basing predictions on well established psychometric scales. Indeed to the best of our knowledge this is the first study to tackle predictions from heterogeneous, asynchronous, longitudinal user generated content.

7.2 A Dataset of Heterogeneous Textual and Mobile phone data

Dataset Design In designing our dataset we wanted to collect real-world user-generated content that could provide information about the spatio-temporal influence on users’ mental well-being. For this purpose, our goal was to combine longitudinal textual sources, such as messages and social media posts, with behavioural data, as manifested by mobility patterns and mobile phone usage. To control for the effect of variable age and stage of life we recruited student participants from the same university in a large cosmopolitan city (New York); unlike [256] the study was not confined to a campus environment. A cohort of 29 students gave us access to their Twitter and Facebook posts, SMS and Facebook messages as well as their mobile phone use data, together with location information and mobile phone sensors, over a period of 4 months each. Data collection was passive, with the exception of on-line submission of psychological tests for well-being (WEMWBS)[235] and affect (PANAS)[257, 54], which students were asked to complete once a day, in the evening. WEMWBS was chosen as a robust, widely used measure of well-being, suitable for the general population and employed by the NHS. Since WEMWBS focusses on positive attributes, we also used PANAS to capture negative emotions. Unlike other work, we did not require any other manual effort from students such as the completion of on-line questionnaires mapping them to personality traits or prompts for self-reported emotional status.

As stated in the previous section, we have only used 19 out of 29 subjects in our experiments in the current chapter, owed to missing data (see the second paragraph of section 7.3.1 for details). Nevertheless, the sample size is in the same magnitude with previous studies within the domain (see Table 3.2). We discuss various issues arising owed to the small sample sizes used in automatic mental health assessment tasks in the next chapter.

Data Collection The data was primarily collected from the Twitter API and two applications (Apps) that were installed for the purpose of the study, on the participants' mobile phones. The first App is DeviceAnalyzer [252], which collects a wide range of time-stamped data, including location and phone usage (e.g. number and duration of calls). SMS data was collected through the NUS SMS collection App², which was configured to retrieve a batch of SMS messages authorised by a participant, as a weekly email. Users were asked to complete psychological scales (mood forms) by logging into a secure webserver, set up for the study. We collected a total of 2436 mood forms, each corresponding to completed PANAS and WEMWBS scales. Facebook data was downloaded by our participants twice during their time on the study and was uploaded to the secure webserver, where the participants could choose the data they wished to share and make available to us. We thus collected 111,270 textual posts and 42GB of DA data spanning the period February 2015-December 2015. Note that participants' time on the study was staggered, with each participant contributing data for 4 months.

Dataset Description The data is heterogeneous by nature and design and asynchronous, with variable temporal granularity, reflecting a real-world scenario and presenting numerous challenges. The most challenges are presented by the DeviceAnalyzer (DA) data, due to their sheer volume and natural redundancy. For example, aggregates are required to represent most DA features (e.g. number of calls, time spent in a location etc.) but choosing the best aggregate and its respective temporal granularity is not straightforward. Moreover, timestamps are presented in epochs, so they had to be converted to absolute values, to be in alignment with those of textual data. We experimented with different methods for aggregation; for the purpose of this study, the decision was made to aggregate DA features at the hour level, by taking mean or cumulative values for the feature within an hourly interval. We selected a subset

²<http://wing.comp.nus.edu.sg:8080/SMSCorpus/contribution.jsp>

(153) of the DA features that can be potentially indicative of user behaviour, as opposed to being related to purely technical aspects of the phone. The former, among others, include: volume of images and SMS messages, physical sensor readings (physical environment and movement), location in terms of longitude and latitude as well as wireless network and data transfer (digital environment), battery level, ringer and other phone settings (user choices). Data is collected anonymously and linked together through user identifiers.

Location and Wi-Fi connection data A further challenge was presented by how to make use of location and Wi-Fi connection data to allow: (a) compatibility with numeric aggregates (b) direct comparisons between different users, who inevitably spend time at different locations with different Wi-Fi connections, with no direct semantic mappings. Our solution to the above was to rank locations and Wi-Fi connections, respectively, according to the time spent in each of them, by each user. Thus we collected the top 10 locations and Wi-Fi connections per user. See also section 7.3.2.

Sensor data There are 15 different sensors of which only accelerometer and light sensor data are provided by 22 of the 29 participants. Each of the two sensors corresponds to 10 different values, including resolution and range of values at a particular time-point.

Textual data The fields associated with each textual instance are the speaker, the raw text, the absolute time stamp, the data source (e.g. Facebook) and the type of text (e.g. message).

Mood forms Obtaining scores for the mood forms is straightforward and based on the scoring instructions associated with each of the two psychological scales.

7.3 Methodology

7.3.1 Data matrix creation and Features

Our goal here is to combine features from both (i) the DeviceAnalyzer (DA) data and (ii) the textual sources (TEXT), in order to train a model that can automatically predict mood scores originating from the three daily mood forms. The latter correspond to the determination of positive affect (“positive”) and negative affect (“negative”), calculated on the basis of the PANAS psychological scale and well-being (“wellbeing”), calculated on the basis of the WEMWBS psychological scale. Those three scores for positive, negative and well-being constitute our target values. Past research has shown a strong correlation between well-being and positive ($r=.71$) and a moderate (negative) correlation between well-being and negative affect ($r=-.54$) [235]. For the purpose of this work we keep the three targets distinct from each other, to aid the interpretability of results.

We had 29 participants on the study who agreed to give us access to both their DA and TEXT data and complete daily mood forms. During the study, two participants switched to iPhones, so they could no longer run DA on their mobile phones. For others, there was missing DA data, where missing data are defined as cases where one or more sources of DA data have no values for longer than a 6 hour period before the completion of a mood form, which was assumed as being most relevant for its completion. TEXT data on the other hand are never considered missing, as the lack of a post is considered to be a choice and a useful indicator of user behaviour. For the purposes of the current chapter, we focused thus on the 19 users for whom we had both DA and TEXT data and no missing data in the 6 hour period prior to the completion of a mood form. This means that from an original set of 2,436 mood forms, each corresponding to three mood score values, several textual posts (Twitter, Facebook, SMS) and several GB of DA data, we make use of 1,438 mood forms and the corresponding features and target values. Thus, for this study, we

used 40,786 textual posts written in English and the corresponding DA data (~ 10 GB). Mood scores consist in scores for well-being, positive and negative affect. Figure 7.1 shows the mean values and the standard deviations for the three mood form scores based on the subjects that were used in our study. The average per-subject score is 25.2, 19.2 and 42.6 for the positive, negative and well-being target respectively. Interestingly, we observe that the average per-subject standard deviation is 5.0, 4.9 and 5.7 for the three targets, pointing to the subjects' affect and well-being fluctuations during the studied period, which makes our task more challenging and shows that simply identifying a subject based on his/her id is not sufficient for predicting his/her mood.

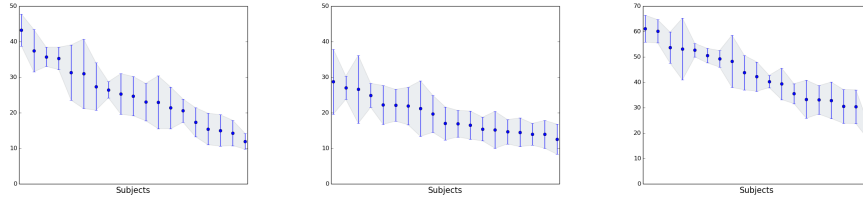


Figure 7.1: Average and standard deviations of the mood form scores obtained by the 19 subjects.

Our textual and DA data points have very different temporal granularity, with hundreds of DA data points in between textual posts. As mood forms are completed every 24 hours (some users being more diligent than others), we decided to extract features within the 24 hour window of a mood form. The underlying assumption is that those features generated by a user during the past day are most likely to have influenced her mood, resulting in the observed mood scores. Thus, given a mood form completed by a certain user at time t , we focused on her past 24 hours before t , in order to extract our features from and aggregate these features in different time windows within the 24 hour period, and, more specifically, into 5 different windows (1, 6, 12, 18 and 24 hours before the completion of a mood form), to allow for an extra level of granularity to the effect of proximity to the mood form timestamp. This process was performed for the DA features that are described in the following section and not for the

TEXT ones, which are only considered at the 24 hour window. This is due to the sparsity of some feature representations of the latter. In future work, we plan to make better use of the temporal granularity of the TEXT features and their interaction with the DA data. In the following, we describe a number of baselines, utilising subsets of the features (7.3.2) and different algorithms, tested under different settings (7.3.3) to establish the most effective approach to combining heterogeneous data for prediction.

7.3.2 Baseline definition

Baseline DA Features Previous work in a controlled user study [256] looked at exploiting features from students’ mobile phone usage within a semester, to predict student academic performance at the end of the semester. While we consider target objectives at much finer grained temporal intervals, we adopt a baseline from mobile phone data (DA) to approximate the ones considered in the StudentLife study [256]. The latter relies on pre-built classifiers (i.e., accelerometer data [140]) to make use of sensor data, such as accelerometer, while we use aggregates of raw data. In our work, we have built classifiers that take into account all data variables, and as such offer more degrees of freedom, to better understand the underlying causes of emotions than studies that consist of disparate pre-built classifiers. Our DA baseline consists of:

- **Calls:** The total number and duration of the calls that a subject has made and received.
- **Locations:** The percentage of time that a subject has spent in her i^{th} preferred location.
- **Wi-Fi:** The percentage of time that a subject has spent while connected to her i^{th} preferred Wi-Fi.
- **Other:** the percentage of time that a user’s mobile: (i) headphones have been “on” (“off”); (ii) screen brightness has been set to “manual”

(“auto”); (iii) airplane mode has been “active” (“inactive”); (iv) ringer mode has been set to “vibrate”, “silent” and “normal”; (v) headset has been “on” (“off”); and (vi) has been disconnected, plugged in a USB port and plugged in AC.

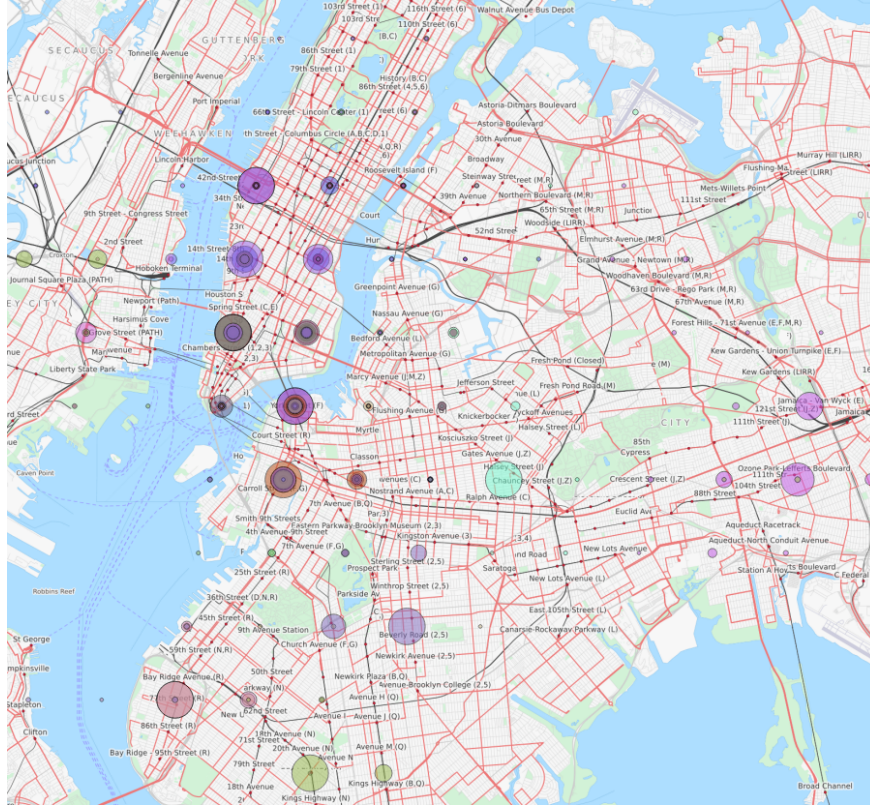


Figure 7.2: Geo-visual projection of the subjects’ visited locations. Each colour indicates a unique user and the size of their spot indicates the number of unique GPS samples at that location.

For locations and Wi-Fi connections we generated features for $i = \{1, \dots, 10\}$, the ten preferred locations and Wi-Fi connections respectively, and an eleventh feature, signaling respectively the total time spent in locations and Wi-Fi access points, other than the top ten. Figure 7.2 shows the projection of the locations visited by the subjects within the city of New York. All DA features were extracted from five different time windows, before the completion of a mood form (1, 6, 12, 18 and 24 hours), leading to 200 DA features per instance. In the case

of missing data for some feature (e.g. missing locations due to disconnections), we filled-in the gaps, by replacing the missing values of a feature with the past 6-hour mean of the same feature for that specific user. For example, if we have no indication of the time spent in particular locations 1 hour prior to the completion of a mood form, we use the 6-hour mean of each location feature from the 6-hour window leading up to the timestamp of the mood form for the user in question. If after this process an instance would still have some missing feature values, we would drop the instance out of our analysis. This resulted in reducing our dataset from 2,436 instances (mood forms completed by 29 users) to 1,438 complete ones, corresponding to 19 different users. Note that while we have sensor data from the phones (accelerometer and light sensor), and accelerometer data were quite predictive in the StudentLife study, we have not used them for the purposes of the current study, due to a large number of missing values exceeding a 6-hour window.

Baseline TEXT Features All the texts (SMS and social media posts/messages) sent by a specific user over the past 24 hours before the completion of a mood form were concatenated in one 24-hour window. Focusing only on the English texts³, the following commonly applied practices were performed: lowercasing, tokenisation [83], replacement of usernames and URLs with placeholders, “usrmnt” and “urlink”, respectively. We extracted the following textual features as potentially relevant to the mental state of users:

- **Ngrams:** We extracted *tfidf* representations of uni- and bi-grams, setting the max (min) document frequency to 99% (1%) and excluding all English stopwords, for noise reduction purposes.
- **Word embeddings:** We used the word embeddings created by [231], which have been used successfully before for the task of sentiment analysis, related to our problem. The unigrams of every text were matched against

³Language detection was performed using <https://pypi.python.org/pypi/langid>

those vectors and seven functions were applied on every dimension of the resulting matrix (mean, median, min, max, stdev, first and third quartile).

- **Lexicons:** We employed several lexicons that have been effectively used in sentiment- or emotion-related works. Those were the Opinion Lexicon [100], NRC Hashtag, NRC Hashtag Emotion [158], Unigram and Bigram NRC Hashtag Sentiment and Sentiment 140 lexicons [259], MaxDiff Twitter Sentiment Lexicon [121], MSOL [159] and AFINN [172]. For lexicons providing binary values (pos/neg), we counted the number of ngrams matching each of the positive and negative classes; for those lexicons with score values, we used the simple counts and the total summation of the corresponding scores from each ngram in the text matched against the lexicons.
- **Topics:** In order to better categorise the content that a subject has shared and to accommodate the sparse representations of the ngrams, we used the word clusters created by Preoţiuc-Pietro et al. [202], which were based on word2vec representations of the most common keywords appearing on Twitter over a 2-month period. We measured the cosine similarity of the unigrams of every textual instance with each one of the 200 word clusters.
- **Other:** We extracted the following features related to the social activity level of a user: the number of SMS messages, Facebook posts, Facebook messages, Facebook images, twitter posts, twitter messages, and the total number of tokens and textual items (messages or posts) in the instance.

7.3.3 Experiments and Models

We applied five regression models, in order to predict each of the three target mood scores separately. All models were tested using 5-fold cross validation using the two sets of features (DA, TEXT) individually and in combination (ALL). Before feeding our features to the regression models, various transformations and normalisation techniques were tested. Those include:

- **The root transformation of the target labels**, often used in regression models to inflate the difference between lower values and stabilise the difference between higher scores⁴.
- **Combinations of:** (a) **normalisation** (linear transformation of feature values to the $[-1, 1]$ range, based on the maximum/minimum value of the feature), (b) **standardisation** (zero mean, unit variance) or (c) **no transformation**.

Those transformations were performed on (i) a per-user basis (so that the feature values of different users become more comparable) and (ii) an overall basis (as a final transformation of all features from different users before applying our models). Notice that in the case of the per-user transformations, the model suffers from the cold-start problem, as it expects to have some past knowledge about the user, in order to predict her mood.

The algorithms that were tested under this setup were Linear Regression, LASSO, Random Forest for regression (RF), Support Vector Regression (SVR) and a multi-kernel SVR approach. The first four algorithms were chosen as widely accepted standards for regression problems, as well as for their diversity (two linear models, one with and one without feature selection, an ensemble of trees, a kernel-based method). Multi-kernel learning (MKL) was proposed in order to allow for a more advanced handling of the different data sources, by jointly learning different kernels, each optimised to a particular data source. For LASSO, different experiments with respect to the alpha parameter were tested (10^{-2} , ..., 10^2); for RF we set the number of estimators to 200, after experimentation; for SVR we have used the Gaussian Kernel with varying kernel width and C values (all combinations of $\{10^{-2}, \dots, 10^2\}$ for both)⁵.

One drawback of SVR is the difficulty to interpret predictions and feature importance. Similarly performing algorithms, such as RF, can provide some in-

⁴We also tried log-transformation but performance was lower.

⁵Python sklearn library (<http://scikit-learn.org/stable/>) was used for the first three models and the Python interface for the Shogun library (<http://www.shogun-toolbox.org>) was used for SVR and MKL.

dication of feature importance in the model learnt, but, when dealing with heterogeneous data sources, data source contribution is a lot less straightforward. For these reasons, we applied the MKL approach [227] which was presented in the previous chapter. Formally, for a training set comprised of instances I and features S partitioned in subgroups $s \in S$, we apply a base kernel k per feature subgroup with some weight w , as follows:

$$f(\mathbf{x}) = \sum_{i \in I} \alpha_i \sum_{s \in S} w_s k_s(\mathbf{x}, \mathbf{x}_i) + b \quad (7.1)$$

where the parameters α_i , the bias term b and the kernel weights are estimated by solving the optimisation problem in Eq. 6.3. We have opted for the L_2 norm to regularise the kernel weights. In order to compare our MKL approach with SVR, we selected one Gaussian kernel per feature set (9 kernels: 4 DA and 5 TEXT, for each of the feature sources defined in 7.3.2) and tuned the width of every kernel and the C parameter performing the same grid search as with SVR. This implies that we have used the same width for all nine kernels in every run. Further kernel selection and parameter optimisation techniques could be used, but those are out of the scope of the current work.

7.4 Evaluation and Results

We have used two standard measures for evaluating our models – the root mean squared error (RMSE, ϵ) and the coefficient of determination (R^2). Those were selected in order to compare both the errors between the different approaches as well as the proportion of the variance that is predictable by them.

Table 7.1 presents the results obtained from our models. We provide separate results of the models for the two cases with respect to the per-user transformation of the features. Only the best transformation combinations are presented per model and the results obtained by Linear Regression are omitted, due to its poor performance. The feature transformation that was used is provided as an

index.

		Positive				Negative				Well-being			
		+User Norm		-User Norm		+User Norm		-User Norm		+User Norm		-User Norm	
		R^2	ϵ	R^2	ϵ	R^2	ϵ	R^2	ϵ	R^2	ϵ	R^2	ϵ
DA	LASSO	n,n .31	8.24	s,s .35	7.99	n,n .11	6.71	s,s .22	6.25	n,n .30	10.51	s,s .35	10.15
	RF	s,s .69	5.55	s,s .64	5.95	s,s .43	5.38	s,s .40	5.49	s,s .75	6.33	s,s .67	7.18
	SVR	n,n .58	6.38	n,n .60	6.27	n,n .35	5.74	n,n .36	5.69	n,n .62	7.80	n,n .62	7.77
	MKL	n,n .61	6.15	n,n .59	6.36	n,n .38	5.60	n,n .33	5.82	n,n .65	7.43	n,n .62	7.80
TEXT	LASSO	n,n .53	6.80	n,n .06	9.59	n,n .23	6.23	n,n .02	7.02	n,n .55	8.46	n,n .10	11.96
	RF	n,n .70	5.42	n,n .13	9.22	n,n .45	5.26	n,n .07	6.85	s,s .74	6.36	s,s .21	11.19
	SVR	n,n .60	6.27	n,n .11	9.31	n,n .32	5.87	n,n .06	6.88	n,n .62	7.72	n,n .19	11.30
	MKL	n,n .62	6.08	n,n .14	9.16	n,n .36	5.69	n,n .06	6.89	n,n .65	7.43	n,n .22	11.12
ALL	LASSO	n,n .49	7.07	n,n .31	8.20	n,n .18	6.41	n,n .20	6.33	n,n .54	8.52	n,n .38	9.92
	RF	n,n .71	5.31	n,n .63	6.00	n,n .46	5.20	n,n .40	5.51	n,n .76	6.23	n,n .68	7.12
	SVR	n,n .60	6.27	n,n .55	6.62	n,n .34	5.76	n,n .31	5.88	n,n .62	7.75	n,n .58	8.17
	MKL	n,n .65	5.84	n,n .61	6.14	n,n .41	5.45	n,n .36	5.67	n,n .68	7.12	n,n .64	7.58

Table 7.1: R^2 root mean squared error (ϵ) of the different models based on the three feature sets (DA, TEXT, ALL) and with respect to the three different ground truth scores (positive, negative, well-being). Values for both setups with respect to the user normalisation (with and without) are presented. The index used in the R^2 column indicates the (i) final and (ii) per-user normalisation of the best-performing setup (n for normalisation, s for standardisation, $-$ for none). Only the final normalisation method (i) is indicated in experiments performed without per-user normalisation.

In terms of comparing the three tasks (predicting each of the targets), our models can successfully capture much of the target variance in their predictions with respect to the well-being target. The lowest errors are observed with respect to the negative target (the comparison with the well-being case in terms of the error is not straight-forward, due to the larger scale that is used in WEMWBS). However, R^2 for this task is considerably lower, pointing to the low variance in each model’s prediction.

The task of user normalisation does not appear to have any significant effect when applied on the DA features for any task, implying that our models trained on DA features are user-independent and can generalise well. However, this is not the case for the TEXT features: for all algorithms and all tasks, the performance drops significantly when no user normalisation is applied. This is an interesting finding, pointing to future work on text-based user modelling, as it provides some evidence that population-wide analyses on mood prediction tasks that do not take it into account can be ineffective.

The comparison between the different algorithms illustrates that RF is the

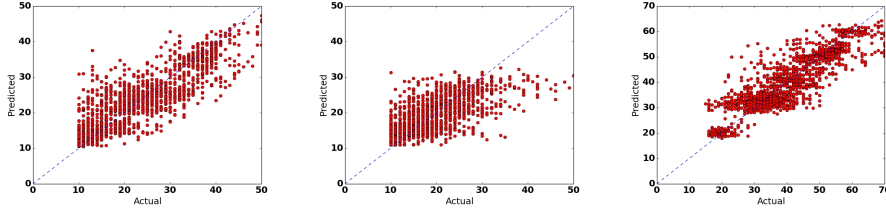


Figure 7.3: Actual VS Predicted charts for the best performing algorithm (RF) on the three targets.

best in most experiments with respect to all target scores, achieving an R^2 of .76 in the best case (predicting the well-being score based on ALL features with user normalisation). To allow comparison with multi-modal affect our ρ scores for RF for {positive, negative, well-being} are {.84, .68, .87} respectively, which is higher than for equivalent multi-modal tasks [89]. The charts in Figure 7.3 illustrate the corresponding predictions graphically. While our MKL does not outperform the RF, it achieves higher accuracy compared to SVR, showing that heterogeneous sources or feature sets can be effectively modelled via multiple kernels with a different weight, depending on their relative impact on the task. Importantly, this improvement comes without any kernel selection or dense parameter optimisation, which can be explored in future work. Also, comparing the results between MKL and RF in the cases without user normalisations shows that MKL is more robust to the cold-start problem for all three targets. This is important, as expecting to have past knowledge from any user is more challenging and resource greedy.

A major advantage of MKL compared to SVR is the interpretation of the feature weights. By comparing the different kernel weights, we can see the contribution of each feature set separately. The bar charts in Figure 7.4 show the weights of each kernel (feature set), as determined by MKL, normalised to sum up to 1. For comparison purposes, we also present the corresponding weights from RF that were extracted by measuring every feature’s importance across the trees and manually mapping those to the MKL’s feature sets. For both the positive and the well-being targets, there are three TEXT feature sets

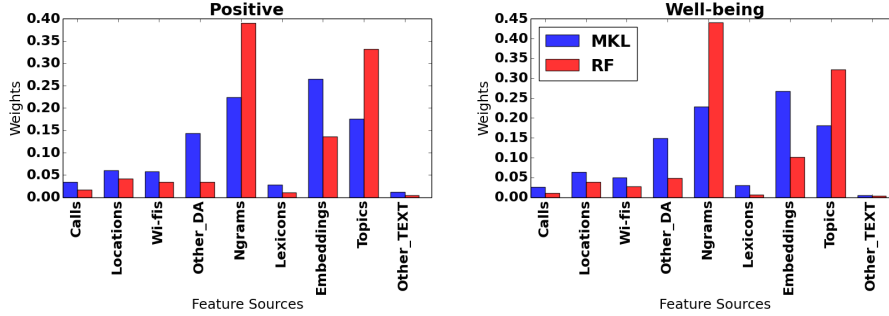


Figure 7.4: Feature set weights in RF (red) and MKL (blue) for the positive and the well-being targets, training on all features in the per-user normalisation approach.

that are preferred by both models (ngrams, word embeddings and topics), albeit with different weights. On the one hand, this points to a possibly weak feature engineering with respect to the DA data. On the other hand, it also explains the difference in accuracy between the two models, which can be highly reduced with further tuning of the MKL kernels and their parameters. Importantly though, the comparison between the feature weights of the two algorithms also explains the relatively small boost in accuracy that RF achieves when the DA features are incorporated in the TEXT-based RF (see Table 7.1), since the algorithm is relying much more on the TEXT features (12.7%, 18.0% and 12.7% of the feature weights come from DA sources with respect to positive, negative and well-being, compared to 29.5%, 28.9% and 28.7% for MKL). This means that MKL has much more potential in making use of heterogeneous sources compared to RF and further tuning of our MKL approach can provide an even more balanced kernel weighting for robustness purposes, while also increasing performance.

7.5 Summary and Conclusion

In the current chapter we have presented our approach towards tackling RQ3 (i.e., nowcasting real-world indices at the micro-level through heterogeneous

user-generated data), focusing on the task of assessing the mental health of a cohort of users in a longitudinal fashion using data derived from their social media and smart phone devices.

In particular, we have presented a new real-world dataset consisting of heterogeneous, longitudinal and asynchronous textual and mobile phone use data. We have investigated different approaches for combining this heterogeneous data for daily predictions of mental health scores at the micro-level. We have expanded the MKL approach presented in Chapter 6 to account for heterogeneous sources, each modelled via a different kernel, and we have compared its performance against competitive baselines, achieving promising results.

However, an important aspect that is a key motivation throughout this Thesis is the ability to build models that can be applied in the real-world. As opposed to Chapters 4–6, where we have worked under a real-world setting, in the current chapter we have not followed such a setup. In specific, in the current chapter, we have divided the instances derived from different users in a randomised, user-agnostic fashion and we have performed evaluation using five-fold cross validation. Such an evaluation cannot guarantee that our model can generalise to new users (thus, that it can be employed in a real-world application). In what follows in Chapter 8, we examine closely issues related to the ability of mental health assessment models to generalise under such a setting, aiming to address the final objective of RQ3 (O3).

CHAPTER 8

Challenges in Micro-level Modelling Using Heterogeneous Data Sources

RQ3/O3c. *Can we apply the micro-level models developed in RQ3, under a real-world setting?*

A key motivation in this Thesis is the real-world deployment of the developed macro- and micro-level modelling approaches. In Chapter 7 though, we proposed a model to assess the mental health of a certain cohort of users, assuming independence between the instances. This implies that we have not tested the model’s ability to generalise over new users, which is crucial in a real-world setting. Furthermore, during evaluation, we provided user-agnostic data as input to our model, but we did not examine whether the model can infer the user’s identity, based on observations of the same user in the training set, and thus provide biased predictions.

In this chapter we take a closer look at ours as well as other state-of-the-art approaches on this task in order to assess their ability to generalise [134, 36, 104, 242]. We demonstrate that under a pragmatic evaluation framework, none of the approaches deliver the reported performances. In fact, we show that current state-of-the-art approaches can barely outperform the most naïve baselines in the real-world setting, posing serious questions not only about their deployment ability, but also about the contribution of the derived features for the mental health assessment task and how to make better use of such data in the future¹.

¹The current chapter is based on [241].

8.1 Introduction

In the previous chapter, we introduced the task of automatically assessing well-being of an individual using data derived from his/her smartphone and social media. Over the latest decade, there has been a growing body of work around this domain, which could possibly revolutionise the mental health assessment process. Most of these studies are longitudinal, where data about individuals is collected over a period of time and predictions of mental health are made over a sliding time window. Having such longitudinal studies is highly desirable, as it can allow fine-grained monitoring of mental health.

However, a crucial question is *what constitutes an appropriate evaluation framework*, in order for such approaches to be employable in a real world setting. Generalisation to previously unobserved users can only be assessed via leave-N-users-out cross-validation setups, where typically, N is equal to one (**LOUOCV**, see Figure 8.1 and Table 8.1). However, due to the small number of subjects that are available, such generalisation is hard to achieve by any approach [134]. Alternatively, personalised models [36, 134] for every individual can be evaluated via a within-subject, leave-N-instances-out cross-validation (for N=1, **LOIOCV**), where an instance for a user u at time i is defined as a $\{X_{ui}, y_{ui}\}$ tuple of $\{\text{features}(u, i), \text{mental-health-score}(u, i)\}$. In a real world setting, a *LOIOCV* model is trained on some user-specific instances, aiming to predict her mental health state at some future time points. Again however, the limited number of instances for every user make such models unable to generalize well. In order to overcome these issues, previous work [25, 73, 103, 104, 221, 242] has combined the instances $\{X_{u_j i}, y_{u_j i}\}$ from different individuals u_j and performed evaluation using randomised cross validation (**MIXED**). While such approaches can attain optimistic performance, the corresponding models fail to generalise to the general population and also fail to ensure effective personalised assessment of the mental health state of a single individual.

In this chapter we demonstrate the challenges that current state-of-the-art

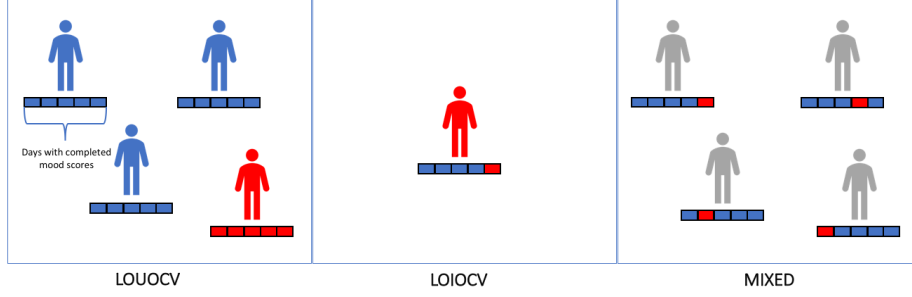


Figure 8.1: The three types of evaluations found in literature. Train instances (or users, in *LOUOCV*) are coloured blue; test instances (users) are coloured red.

	LOUOCV	LOIOCV	MIXED
Real world aim	Build a model m that generalises to a previously unseen user u	Build a personalised model m_u per user u that generalises on u , given some manual input by u	Build a model m that generalises to new instances of a specific pool of previously seen users
Train	$\{\{X_{ui}, y_{ui}\}\}$	$\{\{X_{ui}, y_{ui}\}\}$	$\{\{X_{u_0i}, y_{u_0i}\}, \dots, \{X_{u_ni}, y_{u_ni}\}\}$
Test	$\{X_{ui}, y_{ui}\}$	$\{X_{ui}, y_{ui}\}$	$\{\{X_{u_0i}, y_{u_0i}\}, \dots, \{X_{u_ni}, y_{u_ni}\}\}$
Limits	Few users for training and evaluation	Few instances per user for training and evaluation	Cannot ensure generalisation neither over new users nor in a personalised way

Table 8.1: Summary of the three evaluation frameworks.

models face, when tested in a real-world setting. We work on two longitudinal datasets with four mental health targets, using different features derived from a wide range of heterogeneous sources. Following the state-of-the-art experimental methods and evaluation settings, we achieve very promising results, regardless of the features we employ and the mental health target we aim to predict. However, when tested under a pragmatic setting, the performance of these models drops heavily, *failing to outperform the most naïve – from a modelling perspective – baselines*: majority voting, random classifiers, models trained on the identity of the user, etc. This poses serious questions about the contribution of the features derived from social media, smartphones and sensors for the task of automatically assessing well-being on a longitudinal basis. Our goal in this chapter is to flesh out, study and discuss such limitations through extensive experimentation across multiple settings, and to propose a pragmatic evaluation and model-building framework for future research in this domain.

8.2 Problem Statement

In the current section we describe three major problems identified with respect to model generalisability limitations in previous work. In longitudinal models for mental well-being assessment, a cohort of subjects is monitored over a certain time period (e.g., their social media accounts, smartphones, etc.). During the same time, the subjects are asked to complete a psychometric scale on a longitudinal (e.g., daily) basis. The behaviour of every individual (as revealed through her data) is quantified in order to generate features, which are then used to predict her replies in the psychometric scale throughout time. Ideally, the best way to test the generalising ability of a trained longitudinal model, is to perform a leave-one-user-out cross-validation (*LOUOCV*). This is considered to be a rather difficult task, since the factors affecting one’s mental health vary on an individual basis. An alternative solution is to train personalised models in a within-user, leave-one-instance-out cross validation (*LOIOCV*) manner; while this framework may not be easily generalisable over a wider population, a successful personalised model can guarantee that it can tune in a longitudinal manner to the parameters that affect a person’s well-being. This may require some initial training data for an individual, often consisting of mood forms to be completed manually by the individual that can be used as truth values in combination with other data collected about the individual. The three major limitations we have identified are as follows:

A) Training on past values of the target variable This issue arises when the past N mood scores of a user are used as features to predict his/her next mood score. This is in fact what an autoregressive model of order N does. However, one should not expect major deviations between the mental health state of two consecutive days for an individual [144]. Thus, such approaches are biased towards those past N scores. In a real-world scenario, an application of such an approach would demand the previous N scores of past mood forms, in order to predict the subject’s current mental well-being score, which contrasts

with the goal of automatically assessing well-being (notice that this is different from using the past N *predictions* of these scores, as in many sequential prediction tasks). Most importantly, in this scenario it is difficult to measure the contribution of smartphone-derived features towards the prediction task, as the learning algorithm uses them alongside past mood scores when making a prediction, unless the model is evaluated using target feature ablation (that is, removing the past N scores from the feature set and performing the same evaluation). Furthermore, such models cannot generalise, as they demand manual effort from previously unobserved users. For demonstration purposes, we have followed the experimental setup by LiKamWa et al. [134], which is one of the leading works in the field and follows this paradigm.

B) Inferring test set labels When training personalised models (*LOIOCV*) in a longitudinal study, it is important to make sure that there are no overlapping instances across consecutive time windows. Some past works have extracted features $\{f(t - N), \dots, f(t)\}$ over N days, in order to predict the $score_t$ on day $N + 1$ [36, 134]. While there is no evidence in this field on the history window that one should deploy in order to predict the current $score_t$ of an individual, such approaches are biased if there is (at least) one overlapping day of training/test data. For example, if we train a three-day-historical model on an instance with features $\{f(0), f(1), f(2)\}$ – where $f(x)$ correspond to the features of day x – and ground truth $score_2$ and apply it on the test instance with features $f(1), f(2), f(3)$ and ground truth $score_3$, roughly 67% of our input will be the same, as it is extracted from a very similar time window. Given that the mood scores between two consecutive days are not likely to vary much, we actually risk essentially predicting our training instance. To illustrate this problem we have followed the approach by Canzian and Musolesi [36], as one of the most pioneering works on predicting depression with GPS traces, while following this paradigm.

C) Predicting users instead of mood scores One of the problems with longitudinal studies for assessing well-being is that they seldom are large scale. Instead, they often collect only a few instances from a small group of subjects (so even though there may be thousands of features for each user, typically there is only one truth/target value per day). In order to cope with the lack of substantial data, many works have reused the existing data in several ways – for instance, by mixing all the instances from different subjects, in an attempt to build user-agnostic models in a randomised 10-fold cross-validation framework [25, 104, 103, 242]. While this seems to be a reasonable approach, in reality, such approaches are in danger of “predicting” the user, instead of their mood score. Given that different users exhibit different mental well-being scores on average, this method generates bias in the evaluation: when working on the test set, one can identify who the user is, based on their feature values, even if the user id is not explicitly provided in the training or test set. For instance, as the amount of time spent in the work location on a daily basis does not vary a lot for each user, there exists an expected correlation between the training and test instances for a given single user. Thus, for the identified user, their well-being score can be predicted based on the “average” score observed in the training set for this user. The major problem here is that such approaches cannot guarantee that they will generalise either on a population (*LOUOCV*) or a personalised (*LOIOCV*) manner. For demonstration purposes we have replicated the experimental framework by Tsakalidis et al. [242] (i.e., Chapter 7) and Jaques et al. [103], in order to examine this effect in a regression [242] and a classification task [103].

We now examine closely the issues presented above with specific examples based on leading papers in this area. Due to privacy constraints, we could not have access to the original datasets presented in these works. Nevertheless, the issues raised above and analysed in what follows are of generic nature and dataset-independent. For the rest of this section we present charts derived from the mood scores from the dataset presented in Chapter 7. This dataset consists

of data derived from mobile phones and social media accounts of 30 subjects². The subjects were asked to complete two mood forms on a daily basis: (a) PANAS (leading to two mood scores – “positive” and “negative” – within the [10, 50] range) and (b) WEMWBS (leading to a single “wellbeing” score in the range [14, 70]), which will be presented below.

8.2.1 Training on past values of the target (LOIOCV, LOUOCV)

LiKamWa et al. [134] collected smartphone data from 32 subjects over a period of two months. The subjects were asked to self-report their “pleasure” and “activeness” scores at least four times a day, following a Likert scale (1 to 5), and the average daily scores served as the two targets in the study. The authors aggregated various features on social interactions (e.g., relative number of emails sent to the most frequently interacting contacts) and routine activities (e.g. browsing and location history) derived from the smartphones of the participants. These features were extracted over a period of three days (see next subsection: Inferring Test Labels), accompanied by the most recent scores on activeness and pleasure. The issue that naturally arises is that such a method cannot generalise to new subjects in the *LOUOCV* setup, as it demands their last two days of self-assessed scores. Moreover, in the *LOIOCV* setup, the approach is potentially biased towards the last mood score inputs by the subject, since those are used as an input to the predictive algorithm.

Nevertheless, LiKamWa et al. [134] perform experiments in both *LOIOCV* and *LOUOCV* setups, using Multiple Linear Regression with Sequential Feature Selection (SFS). They find that the personalised models (*LOIOCV*) achieve the lowest error, with an equivalent of 93% accuracy, whereas their *LOUOCV* still yields a good accuracy (66%). However, in the *LOUOCV*, the past two pairs of ground-truth labels of the test user are used as features, creating potential bias and making their approach non-generalisable to completely unobserved users.

²There was a late participant that was not considered in the previous chapter, where the number of subjects was 29.

In order to better examine and argue about the effectiveness of the smartphone-derived features, the same Linear Regression model can be tested without any ground-truth data as input. They show, however, that a naïve model predicting the per-subject average as well as a Linear Regression model trained strictly on the ground-truth features outperforms their *LOUOCV* approach, which yields the question of whether the smartphone-derived features can be used effectively to create a general model that can assess the mental health of unobserved users. Finally, the personalised model in the *LOIOCV* setup clearly outperforms their baselines. However, this model is trained not only on the ground-truth scores, but also over a period of three days predicting the next day; this introduces further potential bias as discussed in the next subsection.

8.2.2 Inferring Test Labels (*LOIOCV*)

Canzian and Musolesi [36] extracted mobility metrics from 28 subjects, in order to predict their depressive state as derived from self-reported PHQ-8 questionnaires [122] that the subjects were completing on a daily basis. A 14-days moving averages filter is first applied to the PHQ-8 scores and then the mean value of the same day (e.g. Monday) is subtracted from the normalised scores in order to avoid cyclic trends of the subject’s mood.

In order to examine the effect of the moving averages filter, we applied it to the three targets (positive, negative, wellbeing) in our working dataset. Fig. 8.2 shows the results of a randomly selected user with respect to the “positive” target. The normalisation results in capturing a longer-term trend of the mental state of the subject and helps in further smoothing the results of mood forms that were completed superficially. However, it also results in losing the dynamic changes in the mood, which are witnessed in the charts before the normalisation was performed and might affect the validation of real-time monitoring of mental health. Furthermore, the target score $score_t$ on a certain day t is now dependent on the past $\{score_{t-14}, \dots, score_{t-1}\}$ scores, making the evaluation in a *LOIOCV* vulnerable to bias if these are used as part of the training set in order to predict

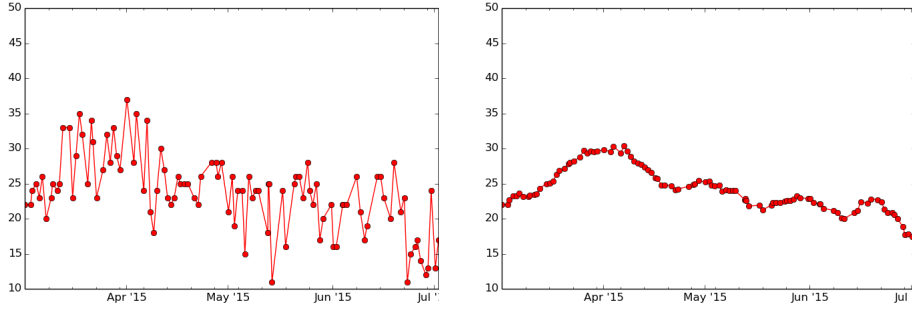


Figure 8.2: Moving averages filter (right) applied to the raw “positive” scores (left) of a randomly selected user. The smoothing effect helps in making the long-term mood trend clearer, in exchange for missing the instant mood score.

$score_t$.

In [36] the normalised PHQ-8 scores are converted into two classes, with the instances deviating more than one standard deviation above the mean score being assigned to the class “1” (or to “0”, otherwise). The features in [36] are then extracted over various time windows (looking at $T_{HIST} = \{0, \dots, 14\}$ days before the completion of a mood form) and model learning and evaluation are performed for every T_{HIST} separately in a per-subject manner, using a *LOIOCV* framework.

What is notable in the results achieved in [36] is that they improve significantly when features are extracted from a wider T_{HIST} window. This could imply that the depressive state of an individual can only be detected with a high accuracy if we look back at her history. However, by training and testing a model on instances whose features are derived from the same days, there is a high risk of overfitting the model to the timestamp of the day in which the mood form was completed. Notice that, in the worst-case scenario in the *LOIOCV* setup, there will be an instance in the train set whose features (e.g., total covered distance) are derived from the 14 days, 13 of which will also be used for the instance in the test set.

While we use the approach by Canzian and Musolesi [36] as an example for the rest of this chapter, a similar approach was also followed in LiKamWa et al. [134] and Bogomolov et al. [25], extracting smartphone-derived features from

the past 2 and 2 to 5 days, respectively.

8.2.3 Predicting Users (LOUOCV)

In Chapter 7 [242], we monitored the behaviour of 19 individuals over four months. The subjects were asked to complete two psychological scales on a daily basis (PANAS [257], WEMWBS [235]), leading to three scores (positive, negative, wellbeing) that were used as the ground-truth; various features from smart-phones (e.g., time spent on the preferred locations) as well as textual features (e.g., ngrams) were extracted over the 24 hours preceding a mood form times-tamp. Four algorithms were applied, in a randomised user-agnostic (*MIXED*) five-fold evaluation setup. We achieve $R^2 = 0.76$ in their best performing setup. However, an interesting case demonstrating the potential user bias is when the models are trained on the textual sources: the highest R^2 is achieved when a Multi-Kernel Learning approach is applied on the wellbeing target (0.22); by normalising the textual features on a per-user basis, the R^2 increases to 0.65 – a pattern which is common for all algorithms targets. This is likely because the users use different vocabularies and thus a normalisation is necessary when working in a mixed users setup, in order to detect their mood. However, in this case there is a danger of overfitting the trained model to the identity of the user, rather than learning a model for recognising their moods. In order to examine this potential, the two different setups (*LOIOCV*, *LOUOCV*) would need to be studied alongside the *MIXED* validation approach, with and without the per-user feature normalisation step.

A similar issue is encountered in Jaques et al. [103] who monitored 68 subjects over a period of a month. Four types of features were extracted from survey and smart devices that subjects were carrying. Self-reported scores on a daily basis served as the ground truth. The authors labelled the instances with the top 30% of all the scores as “happy” and the lowest 30% of the scores as “sad” and randomly separated them into a training, validation and test set, leading to the same user bias issue. This is especially problematic as seen in

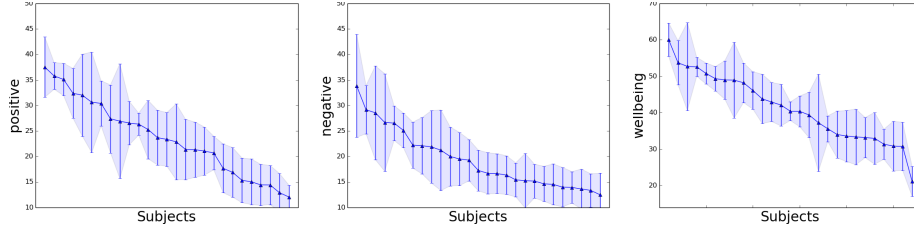


Figure 8.3: Average and standard deviation mood scores (y-axis) on a per-subject basis (x-axis) for the three targets (positive, negative, wellbeing) in our dataset.

Fig. 8.3, since there exist users that exhibit very high and very low mood scores in our dataset. Thus, by selecting instances from the top and bottom scores, one might end up separating users and convert the mood prediction task into a user classification one. Given the personalised nature of the factors that affect our mental health, a more suitable task could have been to try to predict the highest and lowest scores of every individual separately, either in a *LOIOCV* or in a *LOUOCV* setup.

While in this section we use the work presented in Tsakalidis et al. [242] and Jaques et al. [103], it should be noted that a similar experimental setup was followed by Jaques et al. [104], using the median of scores to separate the instances and performing five-fold cross-validation, and by Bogomolov et al. in [25], working on a user-agnostic 10-fold cross-validation from 117 subjects in order to predict their happiness levels, and in [24], for the stress recognition classification task.

8.3 Experiments

8.3.1 Datasets

By definition, the aforementioned issues are feature-, dataset- and target-independent (albeit the magnitude of the effects may vary). To illustrate this, we run a series of experiments employing two datasets, with different feature sources and four different mental health targets.

Dataset 1 We employed the dataset obtained by Tsakalidis et al. [242] (the dataset from our previous chapter), which contains a mix of longitudinal textual and mobile phone usage data for 30 subjects. From a textual perspective, this dataset consists of social media posts and private messages sent by the subjects (see Table 8.2). As opposed to most past work focusing strictly on publicly available data to predict a mental health target, this dataset enables us to study the task at the micro-level, since the vast majority of texts contained therein are private messages ($\sim 94\%$). For our ground truth, we use the {positive, negative, mental well-being} mood scores (in the ranges of [10-50], [10-50], [14-70], respectively) derived from self-assessed psychological scales during the study period.

	messages	posts	images	Overall
facebook	64,221	1,854	447	66,522
SMS	47,043	–	–	47,043
other	132	5,167	0	5,299
Overall	111,396	7,021	447	118,864

Table 8.2: Presentation of *Dataset 1* in numbers by source (rows) and type of item (columns). Note that all messages are private.

Dataset 2 We employed the StudentLife dataset [256], which contains a wealth of information derived from the smartphones of 48 students during a 10-week period. Such information includes samples of the detected activity of the subject, timestamps of detected conversations, audio mode of the smartphone, status of the smartphone (e.g., charging, locked), etc. For our target, we used the self-reported stress levels of the students (range [0-4]), which were provided several times a day. For the approach in LiKamWa et al. [134], we considered the average daily stress level of a student as our ground-truth, as in the original paper; for the rest, we used all of the stress scores and extracted features based on some time interval preceding their completion, as described next, in 8.3.2³.

³For P3, this creates the P2 cross-correlation issue in the *MIXED/LOIOCV* settings. For this reason, we ran the experiments by considering only the last entered score in a given day as our target. We did not witness any major differences that would alter our conclusions.

8.3.2 Task Description

We studied the major issues in the following experimental settings (see Table 8.3):

P1: Using Past Labels : We followed the experimental setting in [134] (see section 8.2.1): we treated our task as a regression problem and used Mean Squared Error (MSE) and classification accuracy⁴ for evaluation. We trained a Linear Regression (LR) model and performed feature selection using Sequential Feature Selection under the *LOIOCV* and *LOUOCV* setups; feature extraction is performed over the previous 3 days preceding the completion of a mood form. For comparison, we use the same baselines as in [134]: Model A always predicts the average mood score for a certain user (**AVG**); Model B predicts the last entered scores (**LAST**); Model C makes a prediction using the LR model trained on the ground-truth features only (**-feat**). We also include Model D, trained on non-target features only (**-mood**) in an unbiased *LOUOCV* setting.

P2: Inferring Test Labels : We followed the experimental setting presented in [36]. We process our ground-truth in the same way as the original paper (see section 8.2.2) and thus treat our task as a binary classification problem. We use an SVM_{RBF} classifier, using grid search for parameter optimisation, and perform evaluation using specificity and sensitivity. We run experiments in the *LOIOCV* and *LOUOCV* settings, performing feature extraction at different time windows ($T_{HIST} = \{1, \dots, 14\}$). In order to better demonstrate the problem that arises here, we use the previous label classifier (**LAST**) and the SVM classifier to which we feed only the mood timestamp as a feature (**DATE**) for comparison. Finally, we replace our features with completely random data and train the same SVM with $T_{HIST} = 14$ by keeping the same ground truth, performing 100 experiments and reporting averages of sensitivity and specificity (**RAND**).

⁴Accuracy is defined in [134] as follows: 5 classes are assumed (e.g., $[0, \dots, 4]$) and the squared error e between the centre of a class halfway towards the next class is calculated (e.g., 0.25). If the squared error of a test instance is smaller than e , then it is considered as having been classified correctly.

P3: Predicting Users : We followed the evaluation settings of two past works (see section 8.2.3), with the only difference being the use of 5-fold CV instead of a train/dev/test split that was used in [103]. The features of every instance are extracted from the past day before the completion of a mood form. In **Experiment 1** we follow the setup in Chapter 7 [242]: we perform 5-fold CV (*MIXED*) using SVM (SVR_{RBF}) and evaluate performance based on R^2 and $RMSE$. We compare the performance when tested under the *LOIOCV*/*LOUOCV* setups, with and without the per-user feature normalisation step. We also compare the performance of the *MIXED* setting, when our model is trained on the one-hot-encoded user id only. In **Experiment 2** we follow the setup in [103]: we label the instances as “high” (“low”), if they belong to the top-30% (bottom-30%) of mood score values (“UNIQ” – for “unique” – setup). We train an SVM classifier in 5-fold CV using accuracy for evaluation and compare performance in the *LOIOCV* and *LOUOCV* settings. In order to further examine user bias, we perform the same experiments, this time by labelling the instances on a per-user basis (“PERS” – for “personalised” – setup), aiming to predict the per-user high/low mood days⁵.

Issue	P1: Training on past labels	P2: Inferring test labels	P3: Predicting users
Setting	LOIOCV, LOUOCV	LOIOCV, LOUOCV	MIXED, LOIOCV, LOUOCV
Task	Regr.	Class.	Regr. (E1); Class. (E2)
Metrics	MSE, accuracy	sensitivity, specificity	R^2 , RMSE (E1); accuracy (E2)
Period	Past 3 days	Past {1,...,14} days	Past day
Model	LR_{sfs}	SVM_{rbf}	SVR_{rbf} ; SVM_{rbf}
Baselines	AVG, LAST, -feat, -mood	LAST, DATE, RAND	model trained on user id

Table 8.3: Summary of experiments. The highlighted settings indicate the settings used in the original papers; “Period” indicates the period before each mood form completion during which the features were extracted.

8.3.3 Features

Dataset 1 For *Dataset 1*, we first defined a “user snippet” as the concatenation of all texts generated by a user within a set time interval, such that the maximum time difference between two consecutive document timestamps is less

⁵In cases where the lowest of the top-30% scores (s) was equal to the highest of the bottom-30% scores, we excluded the instances with score s .

than 20 minutes. We performed some standard noise reduction steps (converted text to lowercase, replaced URLs/user mentions and performed language identification⁶ and tokenisation [83]). Given a mood form and a set of snippets produced by a user before the completion of a mood form, we extracted some commonly used feature sets for every snippet written in English [242], which were used in all experiments. To ensure sufficient data density, we excluded users for whom we had overall fewer than 25 snippets on the days before the completion of the mood form or fewer than 40 mood forms overall, leading to 27 users and 2,368 mood forms. The features that were used in our experiments are the following:

- **duration** of the snippet;
- binary **ngrams** ($n = 1, 2$);
- cosine similarity between the words of the document and the 200 **topics** obtained by [202];
- functions over **word embeddings** dimensions [231] (mean, max, min, median, stdev, 1st/3rd quartile);
- **lexicons** [100, 121, 158, 159, 172, 259]: for lexicons providing binary values (pos/neg), we counted the number of ngrams matching each class and for those with score values, we used the counts and the total summation of the corresponding scores;
- **number** of Facebook posts/messages/images, Twitter posts/messages, SMS, number of tokens/messages/posts in the snippet.

Dataset 2 For *Dataset 2*, we only kept the users that had at least 10 self-reported stress questionnaires, leading to 44 users and 2,146 instances. We extracted the following sets of features:

⁶<https://pypi.python.org/pypi/langid>

- *activity*: percentage of collected samples for each activity (stationary, walking, running, unknown);
- *audio*: percentage of collected inferences for each audio mode (silence, voice, noise, unknown);
- *conversation*: number of conversations detected and their total duration;
- *light*: number of samples and total duration of the time during which the phone was in a dark environment;
- *lock*: number of samples and total duration of the time during which the phone was locked;
- *charge*: number of samples and total duration of the time during which the phone was charging.

Random Features For our *random experiments* used in P2, in Dataset 1 we replaced the text representation of every snippet with random noise ($\mu = 0, \sigma = 1$) of the same feature dimensionality; in Dataset 2, we replaced the actual inferred value of every activity/audio sample with a random inference class; we also replaced each of the detected conversation samples and samples detected in a dark environment/locked/charging, with a random number (<100 , uniformly distributed) indicating the number of pseudo-detected samples.

8.4 Results

8.4.1 P1: Using Past Labels

Table 8.4 presents the results on the basis of the methodology by LiKamWa et al. [134], along with the average scores reported in [134] – note that the range of the mood scores varies on a per-target basis; hence, the reported results of different models should be compared among each other when tested on the *same* target.

	positive		negative		wellbeing		stress		[134]	
	MSE	acc	MSE	acc	MSE	acc	MSE	acc	MSE	acc
LOIOCV	15.96	84.5	11.64	87.1	20.94	89.0	1.07	47.3	0.08	93.0
LOUOCV	36.77	63.4	31.99	68.3	51.08	72.8	0.81	45.4	0.29	66.5
A (AVG)	29.89	71.8	27.80	73.1	41.14	78.9	0.70	51.6	0.24	73.5
B (LAST)	43.44	60.4	38.22	63.2	55.73	71.6	1.15	51.5	0.34	63.0
C (-feat)	33.40	67.2	28.60	72.3	45.66	76.6	0.81	49.8	0.27	70.5
D (-mood)	113.30	30.9	75.27	44.5	138.67	42.5	1.08	44.4	N/A	N/A

Table 8.4: P1: Results following the approach in [134]. The **AVG** baseline outperforms the LR model in *LOUOCV*, consistently. If the features derived from previously self-reported target scores are not used (**-mood**), the performance drops even more.

As in [134], always predicting the average score (**AVG**) for an unseen user performs better than applying a LR model trained on other users in a *LOUOCV* setting. If the same LR model used in *LOUOCV* is trained without using the previously self-reported ground-truth scores (Model D, **-mood**), its performance drops further. This showcases that personalised models are needed for more accurate mental health assessment (note that the **AVG** baseline is, in fact, a personalised baseline) and that there is no evidence that we can employ effective models in real-world applications to predict the mental health of previously unseen individuals, based on this setting.

The accuracy of LR under *LOIOCV* is higher, except for the “stress” target, where the performance is comparable to *LOUOCV* and lower than the **AVG** baseline. However, the problem in *LOIOCV* is the fact that the features are extracted based on the past three days, thus creating a temporal cross-correlation in our input space. If a similar correlation exists in the output space (target), then we end up in danger of overfitting our model to the training examples that are temporally close to the test instance. This type of bias is essentially present if we force a temporal correlation in the output space, as studied next.

8.4.2 P2: Inferring Test Labels

The charts on the left column of Fig. 8.4 show the results by following the *LOIOCV* approach from Canzian and Musolesi [36]. The pattern that these metrics take is consistent and quite similar to the original paper: specificity

remains at high values, while sensitivity increases as we increase the time window from which we extract our features. The charts on the right in Fig. 8.4 show the corresponding results in the *LOUOCV* setting. Here, such a generalisation is not feasible, since the increases in sensitivity are accompanied by sharp drops in the specificity scores. This again demonstrates the difficulty of building a one-size-fit-all model.

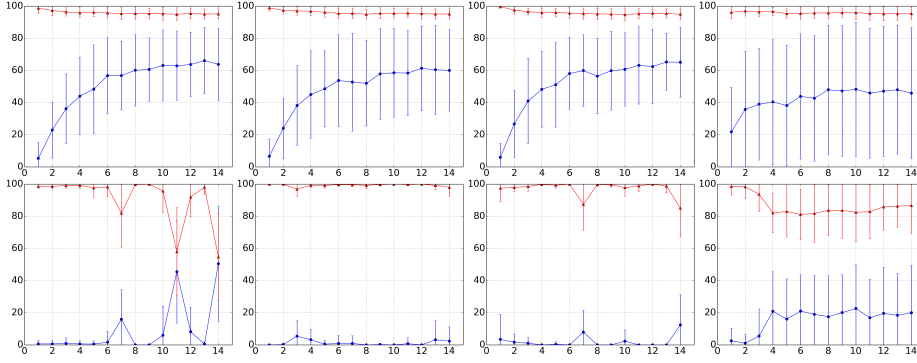


Figure 8.4: P2: Sensitivity/specificity (blue/red) scores over the {positive, negative, wellbeing, stress} targets by training on different time windows on the *LOIOCV* (top) and *LOUOCV* (bottom) setups, similar to [36].

The arising issue though lies in the *LOIOCV* setting. By training and testing on the same days (for $T_{HIST} > 1$), the kernel matrix takes high values for cells which are highly correlated with respect to time, making the evaluation of the contribution of the features difficult. To support this statement, we train the same model under *LOIOCV*, using only on the mood form completion date (Unix epoch) as a feature. The results are very similar to those achieved by training on $T_{HIST} = 14$ (see Table 8.5). We also include the results of another naïve classifier (**LAST**), predicting always the last observed score in the training set, which again achieves similar results. The clearest demonstration of the problem though is by comparing the results of the **RAND** against the **FEAT** classifier, which shows that under the proposed evaluation setup we can achieve similar performance if we replace our inputs with random data, clearly demonstrating the temporal bias that can lead to over-optimistic results, even in the *LOIOCV* setting.

	positive		negative		wellbeing		stress	
	sens	spec	sens	spec	sens	spec	sens	spec
FEAT	64.02	95.23	60.03	95.07	65.06	94.97	45.86	95.32
DATE	59.68	95.92	62.75	95.19	63.29	95.47	46.99	95.17
LAST	67.37	94.12	69.08	94.09	66.05	93.42	58.20	93.83
RAND	64.22	95.17	60.88	95.60	64.87	95.09	45.79	95.41

Table 8.5: P2: Performance (sensitivity/specificity) of the SVM classifier trained over 14 days of smartphone/social media features (FEAT) compared against 3 naïve baselines.

8.4.3 P3: Predicting Users

Experiment 1 Table 8.6 shows the results based on the evaluation setup of our previous chapter, as presented in Tsakalidis et al. [242]. In the *MIXED* cases, the pattern is consistent with [242], indicating that normalising the features on a per-user basis yields better results, when dealing with sparse textual features (positive, negative, wellbeing targets). In Chapter 7 [242], the corresponding average R^2 was 0.51 (from 0.12, without normalisation), which is clearly higher than our 0.39 (from 0.09). The major difference between the two is the number of subjects that were used (27 here, 19 in [242]); while we expect that more subjects would yield better results, the opposite pattern is observed. The explanation of this effect lies within the danger of predicting the user’s identity instead of her mood scores. This is why the per-user normalisation does not have any effect for the stress target, since for that we are using dense features derived from smartphones: the vocabulary used by the subjects for the other targets is more indicative of their identity. In order to further support this statement, we trained the SVR model using only the one-hot encoded user id as a feature, without any textual features. Our results yielded $R^2=\{0.64, 0.50, 0.66\}$ and $RMSE=\{5.50, 5.32, 6.50\}$ for the {positive, negative, wellbeing} targets, clearly demonstrating the user bias in the *MIXED* setting.

The RMSEs in *LOIOCV* are the lowest, since different individuals exhibit different ranges of mental health scores. Nevertheless, R^2 is slightly negative, implying again that the average predictor for a single user provides a better

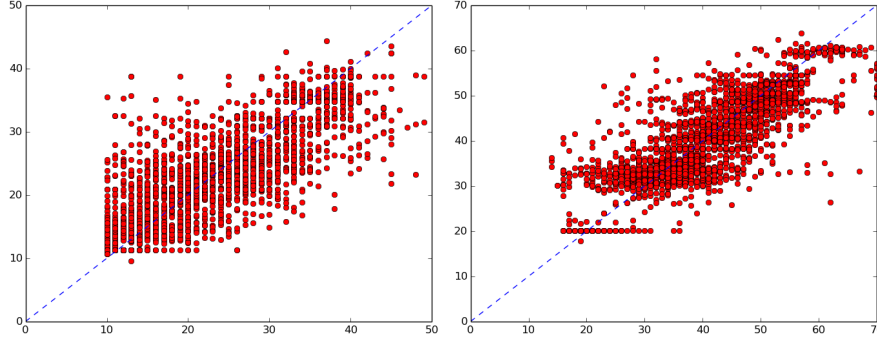


Figure 8.5: P3: Actual vs predicted charts for the “positive” and “wellbeing” targets in *LOIOCV*. The across-subjects R^2 is negative.

estimate for her mental health score. Note that while the predictions across all individuals seem to be very accurate (see Fig. 8.5), by separating them on a per-user basis, we end up with a negative R^2 .

In the unbiased *LOUOCV* setting the results are, again, very poor. The reason for the high differences observed between the three settings is provided by the R^2 formula itself:

$$R^2 = 1 - \left(\sum_i (pred_i - y_i)^2 \right) / \left(\sum_i (y_i - \bar{y})^2 \right)$$

,where f_i is the prediction of the i^{th} instance, y_i is its actual value and \bar{y} is the mean of the actual values in the test set). In the *MIXED* case, we train and test on the same users, while \bar{y} is calculated as the mean of the mood scores across all users, whereas in the *LOIOCV/LOUOCV* cases, \bar{y} is calculated for every user separately. In *MIXED*, by identifying who the user is, we have a rough estimate of her mood score, which is by itself a good predictor, if it is compared with the average predictor across all mood scores of all users. Thus, the effect of the features in this setting cannot be assessed with certainty.

Experiment 2 Table 8.7 displays our results based on Jaques et al. [103] (see section 8.2.3). The average accuracy on the “*UNIQ*” setup is higher by 14% compared to the majority classifier in *MIXED*. The *LOIOCV* setting also yields very promising results (mean accuracy: 81.17%). As in all previous cases, in

	positive		negative		wellbeing		stress	
	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE
MIXED ₊	0.43	6.91	0.25	6.49	0.48	8.04	0.02	1.03
MIXED ₋	0.13	8.50	0.00	7.52	0.13	10.33	0.03	1.03
LOIOCV ₊	-0.03	5.20	-0.04	5.05	-0.03	6.03	-0.08	0.91
LOIOCV ₋	-0.03	5.20	-0.04	5.05	-0.03	6.03	-0.08	0.91
LOUOCV ₊	-4.19	8.98	-1.09	7.24	-4.66	10.61	-0.67	1.01
LOUOCV ₋	-4.38	8.98	-1.41	7.23	-4.62	10.62	-0.69	1.02

Table 8.6: P3: Results following the evaluation setup in [242] (*MIXED*), along with the results obtained in the *LOIOCV* and *LOUOCV* settings with (+) and without (-) per-user input normalisation.

	positive		negative		wellbeing		stress	
	UNIQ	PERS	UNIQ	PERS	UNIQ	PERS	UNIQ	PERS
MIXED	65.69	51.54	60.68	55.79	68.14	51.00	61.75	56.44
LOIOCV	78.22	51.79	84.86	53.63	88.06	52.89	73.54	55.35
LOUOCV	47.36	50.74	42.41	52.45	45.57	50.10	49.77	55.11

Table 8.7: P3: Accuracy by following the evaluation setup in [103] (*MIXED*), along with the results obtained in *LOIOCV* & *LOUOCV*.

LOUOCV our models fail to outperform the majority classifier. A closer look at the *LOIOCV* and *MIXED* results though reveals the user bias issue that is responsible for the high accuracy. For example, 33% of the users had all of their “positive” scores binned into one class, as these subjects were exhibiting higher (or lower) mental health scores throughout the experiment, whereas another 33% of the subjects had 85% of their instances classified into one class. By recognising the user, we can achieve high accuracy in the *MIXED* setting; in the *LOIOCV*, the majority classifier can also achieve at least 85% accuracy for 18/27 users. This also explains the higher accuracy in the {positive, negative, wellbeing} targets under *MIXED*: in these cases we are using textual features, which are more indicative of the user, to predict a class label, which is based on a wider range of values ([10-50], [14-70]) compared to the “stress” target ([0-4]), thus allowing the users to have wider differences from each other in terms of their average mental health scores.

In the “*PERS*” setup, we removed the user bias, by separating the two classes on a per-user basis. The results now drop heavily even in the two previously well-performing settings and can barely outperform the majority classifier. Note

that the task in Experiment 2 is relatively easier, since we are trying to classify instances into two classes which are well-distinguished from each other from a psychological point of view. However, by removing the user bias, the contribution of the user-generated features to this task becomes once again unclear.

8.5 Proposal for Future Directions

Our results emphasize the difficulty of automatically predicting individuals' mental health scores in a real-world setting and demonstrate the dangers due to flaws in the experimental setup. Our findings do not imply that the presented issues will manifest themselves to the same degree in different datasets – e.g., the danger of predicting the user in the *MIXED* setting is higher when using the texts of 27 users rather than sensor-based features of more users [24, 25, 103, 221]. Nevertheless, it is crucial to establish appropriate evaluation settings to avoid providing false alarms to users, if our aim is to build systems that can be deployed in practice. To this end, we propose model building and evaluation under the following:

- **LOUCV**: By definition, training should be performed strictly on features and target data derived from a sample of users and tested on a completely new user, since using target data from the unseen user as features violates the independence hypothesis. A model trained in this setting should achieve consistently better results on the unseen user compared to the naïve (from a modelling perspective) model that always predicts his/her average score.
- **LOIOCV**: By definition, the models trained under this setting should not violate the iid hypothesis. We have demonstrated that the temporal dependence between instances in the train and test set can provide over-optimistic results. A model trained on this setting should consistently outperform naïve, yet competitive, baseline methods, such as the last-entered mood score predictor, the user's average mood predictor and the

auto-regressive model.

Models that can be effectively applied in any of the above settings could revolutionise the mental health assessment process while providing us in an unbiased setting with great insights on the types of behaviour that affect our mental well-being. On the other hand, positive results in the *MIXED* setting cannot guarantee model performance in a real-world setting in either *LOUOCV* or *LOIOCV*, even if they are compared against the user average baseline [61].

Building a generic model that can fit any subject is a rather difficult task, at least when dealing with a small number of individuals [149] and hence many past approaches have instead opted to train personalised models [177, 134, 36]. The main problem in the *LOUOCV* setting is that most machine learning algorithms assume that the features in the training and test domains (in our case, users) follow the same distribution, which is not the case in many applications, including the mental health assessment task, since people express different behavioural patterns and different moods. Techniques from the generic domains of personalisation and **transfer learning** [180] can provide significant help. Such methods aim to adapt to a previously unobserved domain, by using knowledge obtained from observed instances, typically by selecting either instances or features from the known domains and transferring the knowledge obtained from them to the unobserved domain.

A drawback of applying a transfer learning model for this task is that such methods assume that models from the training domains (users) have been successfully learned. However, we have demonstrated that this is not the case in this domain and with datasets of the type currently used by the state-of-the-art, since all of our *LOIOCV* experiments provided negative results. Better feature engineering through **latent feature representations** might be proven to be of crucial importance. While different users exhibit different behaviours, these behaviours might follow similar patterns in a latent space. Such representations have seen great success over the last years in the fields of natural language processing [153] and have been extended to multi-modal [131] and temporal mod-

elling [205, 70], aiming to capture latent similarities between seemingly diverse concepts and represent every feature based on its context. For example, NLP models, such as word2vec, can easily be extended by considering multi-modal characteristics (e.g., current location) as part of a word’s context for a particular user, whereas the same also holds for the non-textual modalities. The resulting representations have the potential to model the user behaviour more accurately, by taking advantage of this latent mapping.

Another aspect of the problem that many past works have ignored is the **fixed characteristics** of the individuals under study, such as their demographic data. Such data can serve as an important source of information, especially in the *LOUOCV* setting, and their successful incorporation in stand-alone user classification tasks has recently been demonstrated by Benton et al. [18] in the mental health domain. However, for an effective incorporation of such information, micro-level data from more users than the current studies are using are needed: working on **larger datasets** will help in building more robust approaches and test their generalisation, since small-scale studies (either with respect to number of participants or duration of time, as in [217] and [101]), are more vulnerable to user bias. Indeed, the number of instances used is a major difference between the stand alone user/text classification tasks, in which the digital media features are successfully integrated, and the longitudinal mental health studies, in which we showed that there are limitations with respect to their deployment. Finally, clear **dataset description** and **problem statement** will provide clearer insights in realising the challenges in the respective tasks, while establishing naive, yet competitive, **baselines** suitable for the problem under study is of vital importance [61] in order to evaluate the mental health predictors properly and avoid creating false alarms in a real world setting.

8.6 Summary and Conclusion

Assessing mental health with digital media is a task which could have great impact on psychological assessments, monitoring of mental well-being and personalised health. In the current chapter, we have followed past experimental settings to evaluate the contribution of various features to the task of automatically predicting different mental health indices of an individual. We find that under an unbiased, real-world setting, the performance of state-of-the-art models drops significantly, making the contribution of the features impossible to assess. Crucially, this holds for both cases of creating a model that can be applied in previously unobserved users (*LOUOCV*) and a personalised model that is learned for every user individually (*LOIOCV*).

Our major goal for the future is to achieve positive results in the *LOUOCV* setting. To overcome the problem of having only few instances from a diversely behaving small group of subjects, transfer learning techniques on latent feature representations could be beneficial. A successful model in this setting would not only provide us with insights on what types of behaviour affect mental state, but could also be employed in a real-world system without the danger of providing false alarms to its users.

CHAPTER 9

Proposal for Future Directions in Micro-level Modelling Using Heterogeneous Data Sources

In the current chapter we present our first steps towards tackling the serious micro-level issues that were presented in Chapter 8, aiming to provide a real-world solution to the task of assessing mental health using digital media. Given that individuals' behaviour vary on a per-user basis, our aim in this chapter is to propose an approach to bridge this gap in order to build user representations that might be linked to their mood in a latent space. We develop an approach on building such “*behavioural embeddings*” and provide proposals for future work in this domain.

9.1 Introduction

A key problem in the task of predicting micro-level mental health indices on a longitudinal basis is the fact that both the digital traces (e.g., language used in social media, locations visited, etc.) and the mental health indices vary on an individual basis. Furthermore, we have demonstrated in the previous chapter that, due to the small number of instances per user, we cannot build effective personalised models under *LOIOCV* that can generalise to a single individual. Hence, the need for building better user and feature representations for our task is important for creating models that can generalise under both *LOUOCV* and *LOIOCV*. For a recap on the concepts of *LOUOCV* and *LOIOCV*, the reader is pointed to Figure 8.1 and Table 8.1.

In the current chapter we propose a method for building such user representations based on the data derived from his/her smart device. Given the raw data derived from the smart devices of some users, we build latent representations of every user and modality, transforming the raw data input (e.g., a one-hot encoded wifi id) into a vector that represents it based on the context that it is usually used (e.g., based on the time that it is used by a certain user in a specific location). For this purpose, we adjust the **word2vec** approach by Mikolov et al. [153] on building efficient word representations in the field of Natural Language Processing. We provide qualitative evidence that such an approach can be employed for building meaningful representations of a user’s behaviour and propose future directions for utilising such representations for the task of assessing mental health using digital media.

9.2 Methods

In the current section we provide details on the approach we have followed in order to build latent feature representations (*“behavioural embeddings”*) that capture the behaviour of individuals, as derived from their digital traces. In particular, we assume representations of users $u_i \in U$ over time as:

$$\{\{x_{u_0}^{(t_0)}, \dots, x_{u_0}^{(t_N)}\}, \dots, \{x_{u_U}^{(t_0)}, \dots, x_{u_U}^{(t_M)}\}\}.$$

Each vector $x_u^{(t)}$ represents the one-hot-encoded features that are derived from user's u smart phone within a certain hour t . For example, within a certain hour t , a user might be in a specific location, while he/she receives a phone call and has his/her phone in silent mode. Such features are represented as a one-hot-encoded vector $x_u^{(t)}$ for this user within the specified hour. Our goal is to create behavioural embeddings based on this raw input space, leading to a lower-dimension and context-based feature space. This latent space has the potential to bring closer seemingly diverse input features, based on the contexts they appear in, which in turn might be related to the users' micro-level indices universally.

In what follows, we provide details on our approach for creating the behavioural embeddings. We work on the dataset that was introduced in the previous chapters [242] (see Chapter 7), using only the smartphone features (i.e., we do not use any textual features), as our first step towards latent user modelling.

9.2.1 Behavioural Embeddings: mobile2vec

We follow an approach based on `word2vec`, which was first introduced by Mikolov et al. [153], aiming to build low-dimensional representations of words, based on the context they appear in. As presented in Chapter 2, `word2vec` defines a classification task, which comes in two flavours: (a) in the Skip-gram model we aim at predicting a certain target word based on its context; (b) in continuous-bag-of-words (CBOW) approach, we aim at using a certain word in order to predict the context in which it normally appears. The process starts by scanning through a large collection of documents and creating $\{\text{context}, \text{target}\}$ pairs of the (one-hot-encoded) words, where every word serves as a “target” and its “context” is defined by the words that fall within a distance based on

a pre-defined window. For example, based on the sentence “*the cat sat on the mat*” we can define the following pairs of {target, context}, assuming a window of size 2:

- {the, [cat, sat]}
- {cat, [the, sat, on]}
- {sat, [the, cat, on, the]}
- {on, [cat, sat, the, mat]}
- {the, [sat, on, mat]}
- {mat, [on, the]}

We adjust this setup for building latent representations of smartphone features (**mobile2vec**). We consider the one-hot-encoded features (e.g., locations, wifi, calls, etc.) within a given hour for a user as our sentence. For a recap on what types of features have collected in our dataset, which we will be using in our modelling, the reader is pointed to section 7.2. We then form all possible {target, context} pairs within an hour; in other words, we do not define a window around the target feature, since our smartphone features are occurring at the same time (e.g., a user is at a specific location and connected to a specific wifi, while talking on the phone) and not sequentially, as in the case of the words within a sentence. Finally, in a similar fashion to **word2vec**, we build a one-hidden-layer feed-forward neural network, aiming to maximise the following objective function:

$$J = \log p_w(D = 1|f, h) + \sum_{\tilde{f} \sim Q_{\text{noise}}} \left[\log p_w(D = 0|\tilde{f}, h) \right]. \quad (9.1)$$

In Eq. 9.1, f denotes a particular feature (e.g., the presence of an outgoing phone call within a certain hour), h is the rest of the contextual features (e.g., location, wifi, etc.), \tilde{f} denotes the negative examples (other than f) and k is the

pre-defined number of negative examples that we use in our modelling. Finally, p_w is the logistic regression probability, given the weights w of the model.

9.3 Training mobile2vec

First, let us define the mobile “sentences” used in our modelling. We assume that the behaviour of a user, as sensed from his/her smart phone, within a certain hour, is a single sentence. This links with the work in Chapter 7, so we assume each instance of a user’s feature values corresponds to a sentence. We extracted 40,723 such sentences in our dataset (that is, hour intervals across all users). The “words” in this sentence are indicated in our modelling by the following (one-hot-encoded) attributes:

- **User id:** a unique id per user.
- **Date:** the month, day and hour of the day.
- **Location:** the location(s) in which the user is at the specified hour.
- **Wi-fi:** the wi-fi(s) that the user is connected to at the specified hour.
- **Calls:** presence or absence of calls made and received.
- **Multimedia:** creation or deletion of images and videos during the specified hour.
- **Airplane mode:** indicating the airplane mode (“on” and/or “off”) during the specified hour.
- **Headset:** presence or absence of headset.
- **Ringer mode:** the ringer mode of the phone (“normal”, “silent”, “vibrate”) during the specified hour.
- **Power:** mode of power connection (“disconnected”, “AC”, “usb”) during the specified hour.

Number of epochs	[5, 10, 15, 20, 25, 50, 100]
Number of negative samples	[1, ..., 10]
Latent space dimensionality	[5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

Table 9.1: Parameters used in `mobile2vec` training process.

- **Brightness:** mode of the screen brightness (“auto”, “manual”) during the specified hour.
- **HF:** the mode of the headphones (“locked”, “unlocked”) during the specified hour.

The selection of these “words” was based on available information we had in our dataset. However, the list can be further expanded if more information is available. The average number of words per sentence in our dataset is 8.33. After generating the sentences in a per-hour and per-user basis, we can create examples of $\{\text{target}, \text{context}\}$ pairs by making all possible combinations of the attributes within a specified hour. We use these examples as input to our model and we train with different parameters, as presented in Table 9.1. After training, each feature is represented by an N –dimensional vector, as indicated by the pre-defined latent space dimensionality in Table 9.1.

Fig. 9.1 demonstrates the loss scores at every 500 iterations, with one chart per dimensionality of the resulting representations and different lines in each chart corresponding to a different number of negative samples used in our modelling. We observe that our model manages to reduce the loss score rapidly over the first iterations, owed to the small number of sentences used in our modelling. This is a highly desirable property, since it indicates that our behavioural embeddings can be trained very fast. However, to assess their effectiveness on capturing the context of the raw features, we need to examine the resulting representations more closely, as studied next.

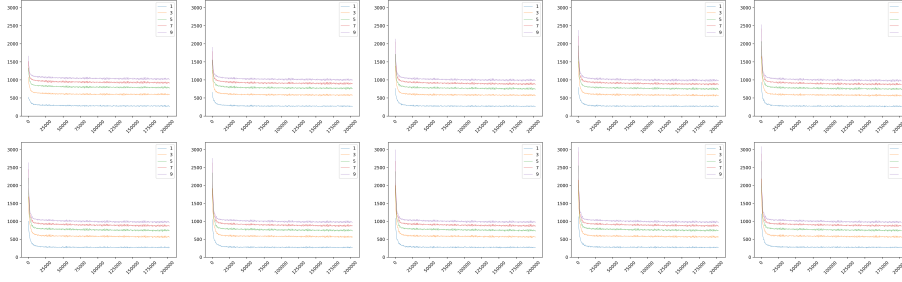


Figure 9.1: Loss per 500 iterations, using different number of negative samples (lines) and latent space dimensionality (above: [10, ..., 50]; below: [60, ..., 100]).

9.4 Empirical Validation

In this section, we aim at visualising the behavioural embeddings, using t-SNE [145] and measuring the distance between semantically related features. We expect that the semantically related features (e.g., all of the locations, all wi-fis, etc.) will have a low distance score between each other in their respective latent space. We explore the effect of different number of (*a*) epochs and (*b*) dimensionality of the resulting vectors, when training with one negative example.

9.4.1 Number of epochs

Figure 9.2 shows the resulting representations, when training with different number of epochs (5, 20, 50, 100) and one negative example. What is clearly observable is the fact that training for a larger number of epochs results into better, from a semantic point of view, representations. To further support this statement, we calculate the euclidean distance between each of the semantically related feature sets. By “semantically related” we mean feature categories such as {“location0”, ..., “location9”}, {“wifi0”, ..., “wifi9”}, etc. We expect that, as the number of epochs increases, the euclidean distance between semantically related features will be reduced, leading to more semantically meaningful feature representations.

In Figure 9.5, we project the average euclidean distance between the feature categories, per epoch (e.g., the average euclidean distance between all the loca-

Proposal for Future Directions in Micro-level Modelling Using Heterogeneous Data Sources

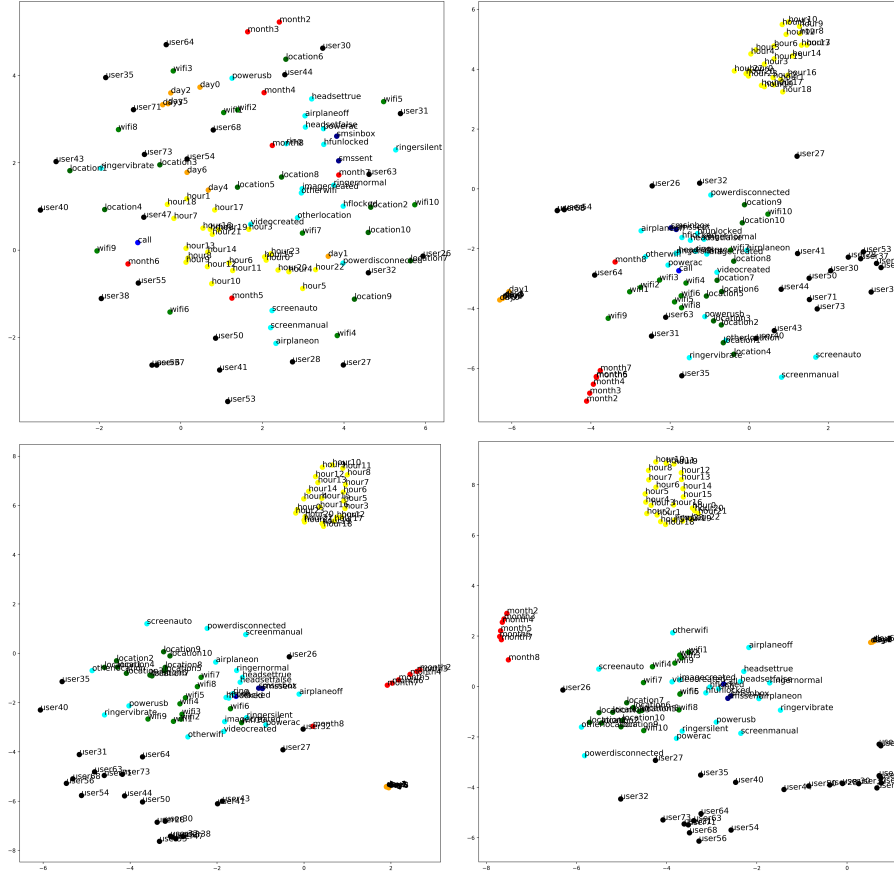


Figure 9.2: Visualisation of the (20-dim) embeddings, using different number of epochs in our training (5, 20, 50, 100).

tion features, per epoch) and using different latent feature dimensionality for our embeddings. In all cases, the distance is reduced during the first 10-20 epochs. However, for low latent feature dimensionality (i.e., < 40), the semantically related features start deviating again when training for a larger number of epochs – an overfitting effect which is probably owed to the small dimensionality of the resulting representations. On the other hand, by increasing the dimensionality of the representations, the corresponding distance increases (note that the scale in the y-axis is different across the charts in Figure 9.5). We examine this more closely in what follows.

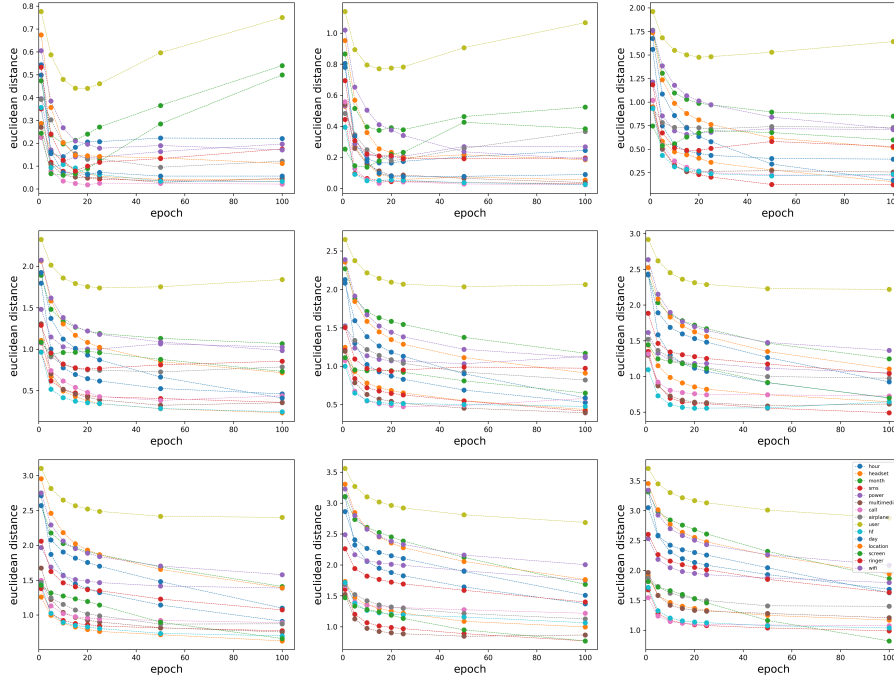


Figure 9.3: Average euclidean distance between semantically related feature categories, per epoch. Different charts correspond to different latent feature dimensionality (5, 10, 30, 40, 50, 60, 70, 90, 100).

9.4.2 Dimensionality of Embeddings

The scatter plots in Figure 9.4 present the resulting embeddings after t-SNE [145], for different latent dimensionality sizes and by using one negative example during training. The plots provide further empirical evidence to our previous note: by increasing the dimensionality of the resulting feature representations, our model fails to group together the semantically related features. To examine this closer, we plot the average euclidean distance between the feature categories as before, this time on a per-epoch basis. Figure 9.5 clearly demonstrates that, regardless of the number of epochs used to train our model, a higher feature dimensionality yields poorer feature representations.

Overall, our empirical findings suggest that a low latent feature dimensionality yields better – from a semantic point of view – representations, even after training our model for a small number of epochs. This is important, since it

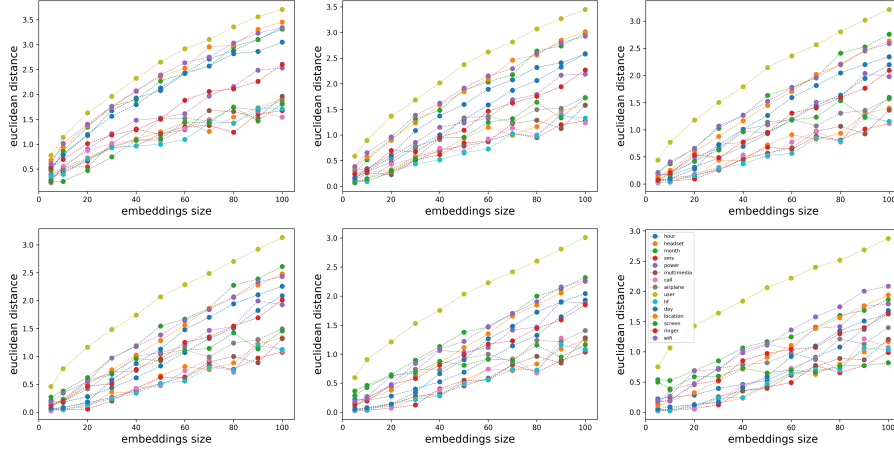


Figure 9.5: Average euclidean distance between semantically related feature categories, per latent feature dimensionality. Different charts correspond to different epochs used during training (1, 5, 15, 25, 50, 100).

9.5 Discussion

Our empirical analysis points that we can build latent and low-dimensional representations of smartphone-derived features, able to capture the context of every feature across different users. Incorporating the resulting feature vectors in a machine learning algorithm that aims to assess the mental health of an individual can be performed in several ways. For example, in our NLP-related modelling in Chapters 4, 6 and 7, we have represented a single document as the average across all dimensions of the words comprising it – a common practice for modelling documents of short length [231, 238]. Similarly, we can extract the average of all feature dimensions over the last day preceding the completion of the mood form of a user. However, this would result into losing the temporal dimension of our task. To accommodate that, we can use temporally sensitive models, such as MCKL (see Chapter 6) or neural RNN-based architectures, operating on the per-hour time-steps.

This lack of the temporal component is present even in the generation process of our latent representations: we have first separated the features based on the hour interval that they appear at and then we have formed pairs of {context,

target} based on co-occurring features. This implies that we fail to capture the similarities between the users’ behaviour on a temporal basis. For example, a user might be at location_0 at 13:00pm and at location_1 at 15:00pm, on a daily basis. Despite that this pattern might be rather frequent, our {context, target} pairs will never have the two locations grouped together in this form. While working with temporally sensitive models might help into capturing the temporal dimension, simple feature-aggregate models will essentially fail to do so. To accommodate that, we can instead generate latent representations that also account for the temporal/sequential nature of the input features. Future directions for this include RNN-based modelling [191] and Tensor Decomposition approaches [223].

Finally, in order to assess the effectiveness of the resulted feature representations for the task of assessing mental health (or any other micro-level longitudinal target), the evaluation should again be performed in an unbiased *LOUOCV* fashion – note that the danger of “predicting the user” in the *MIXED* setting still exists under the new representations (see Chapter 8).

9.6 Summary and Conclusion

In this chapter, we have presented our first steps towards generating latent representations of features derived from smart devices, in an attempt to represent the raw smart phone logs produced by a set of users in a more semantically meaningful way. Our findings suggest that we can generate such “behavioural embeddings” with very little training, resulting into transforming every one-hot-encoded feature in a lower-dimensional and more context-based representation across different users, thus bridging between their diverse behaviours. Such representations can be employed in future work for the task of assessing mental health that was introduced in Chapter 7 in order to tackle the serious issues of current state-of-the-art approaches, which were demonstrated in Chapter 8. Finally, we have proposed alternative ways of generating such latent represen-

tations, taking into account the temporal dimension of the features, which is often ignored in mental health assessment tasks.

Part IV

Conclusion

CHAPTER 10

Conclusions

The wide adoption of social media and smart devices has resulted into unprecedented volumes of user-generated streams of data, offering the opportunity to researchers to use them in order to analyse and quantify real-world indices, in a longitudinal fashion. In this Thesis, we presented different approaches on monitoring such real-world indices in different levels of granularity, by leveraging user-generated textual and heterogeneous data sources. Through rigorous evaluation under a real-world and real-time setup, we have demonstrated the ability of the proposed methods to be employed under such a setting, showcasing their effectiveness against various baseline models in different tasks and highlighting their limitations. In this concluding chapter, we summarise the key findings and propose directions for future work within the domain of nowcasting user behaviour using social media and smart devices in a longitudinal fashion.

10.1 Main Findings

In the current section we summarise our main findings with respect to each of the research questions set up in the introduction.

10.1.1 Preliminary (document-level) Analysis

In Chapter 4, we presented different approaches on generating NLP resources for opinion mining tasks over user-generated content. Working in the – under-resourced – Greek language, we have generated a manually annotated sentiment and emotion lexicon, comprised of 32K word entries, two large-scale sentiment lexicons, generated in a semi-supervised fashion, and word embeddings, trained on a large corpus of 14M documents. We have showcased the benefit of incorporating our resources in traditional models under different opinion mining tasks. In particular, we highlight the following contributions:

- In the across-domain sentiment analysis task, we achieve a 24.9% relative improvement by using our resources (lexicons, word embeddings) compared to using the standard ngram representations. This is highly important, since having an annotated dataset for every existing domain is impossible in a real-world setting.
- In the in-domain sentiment analysis task, we achieve a relative boost in performance of 2.7–5.6% on average (across different datasets) by incorporating our resources, compared to ngrams.
- Our resources have also been effectively incorporated in the emotion detection task, where our word embeddings achieve the best results compared to several other baseline features, across different emotions.

Finally, by making our resources publicly available, along with two manually annotated datasets, we encourage researchers and organisations to employ them in their tasks, aiming to develop approaches that can monitor aggregated opinions of an online population at the macro-level.

10.1.2 RQ1: Macro-level Monitoring Using Social Media

RQ1. *Can we use data streams from social media in order to nowcast real-world indices on the macro-level?*

In Chapter 5, we focused on mining online data streams from social media, in order to nowcast macro-level indices. Focusing on the election prediction task as our case study, we make the following contributions:

- We propose time-series-based models that can leverage large-scale social media posts, in order to nowcast political indices at the macro-level. While our focus was based on the political domain, the proposed methods can easily be adapted to other urban or social macro-level indices.
- To the best of our knowledge, we are the first to publish the electoral forecasts, effectively addressing **C3** challenge (“working under a real-world setting”). Furthermore, we have employed our approach in three electoral cases, with consistent results.
- We have showcased that by incorporating data streams from social media, we can achieve significantly better performance for the task of nowcasting macro-level political indicators, compared to traditional poll-based prediction models.

10.1.3 RQ2: Micro-level Monitoring Using Social Media

RQ2. *Can we use data streams from a specific group of social media users in order to nowcast their real-world indices on the micro-level?*

To address **RQ2**, we have worked on the task of nowcasting the voting intention of social media users. Working under a real-world and real-time setting in this task, we make the following contributions:

- We have presented the first systematic study on nowcasting voting intention of user over time, during a sudden and major political event.
- We have presented a novel semi-supervised approach based on multiple kernel learning of heterogeneous and asynchronous information sources, where every temporally-sensitive information source is modelled through convolution kernels.
- We have demonstrated the effectiveness and robustness of our approach against various baselines for the task of predicting the voting intention of social media users over time. We achieve an almost 20% increase in F-score compared to the best performing baseline which is trained on textual feature aggregates.
- We have provided empirical evidence of the importance of temporal modelling of textual and network-related information, through a thorough qualitative analysis.

Importantly, despite the fact that our focus was paid on a particular case study, the same method can easily be adjusted to other NLP tasks consisting of multiple temporally sensitive information sources.

10.1.4 **RQ3: Micro-level Monitoring Using Heterogeneous Sources**

RQ3. *Can we use asynchronous and heterogeneous data streams from a specific group of users in order to nowcast their temporally sensitive real-world indices on the micro-level?*

To address our final two research questions, we have worked on a different task, in order to also account for a time-varying target. With respect to **RQ3**, we make the following contributions:

- We present the first – to the best of our knowledge – study on using textual and smartphone data to assess mental health indices at the micro-level.
- We apply a regression variant of the multiple kernel learning approach developed in **RQ2** for our task, demonstrating its effectiveness on capturing the mental health indices of our users.
- We provide qualitative evidence on the importance of different features, showcasing that our proposed method makes better use of complementary information sources, compared to baseline models.

Despite the promising results and the effectiveness of our approach, as demonstrated in Chapter 7 of this Thesis, we then alter our experimental and evaluation setup, in order to address our **C3** challenge, corresponding to our **RQ3/O3c** objective:

RQ3/O3c. *Can we apply the micro-level models developed in RQ3, under a real-world setting?*

To this end, we make the following important contributions for the task of assessing mental health using social media and smart devices:

- After a thorough literature review (see section 3.4), in Chapter 8 we present three major problematic issues that apply to almost all past work in this domain. All these issues are derived owed to a non-realistic experimental and evaluation setup.
- Working on different datasets, input features and mental health targets on the micro-level, we follow state-of-the-art approaches that encounter

one or more of the three major issues. We empirically demonstrate that past work build models that are not applicable under a real-world setting. Furthermore, we show that the contribution of the extracted features for the task of assessing mental health is, in fact, in question.

- We propose future directions, aiming to build robust and effective approaches for this task, highlighting the importance of their ability to be employed under a real-world setting.

Our findings demonstrate that these serious flaws in experimental and evaluation design might have affected a large proportion of past studies, thus pointing to the need of establishing appropriate evaluation settings and re-examining the conclusions reached in past work.

One of the issues that arise when working with datasets comprised by a small number of users for the task of assessing their mental health through their online/digital behaviour, is the fact that this behaviour can be quite diverse across the different users. Training a model on such diverse user input features for the task of assessing a new user’s mental health, whose input features again differ from the observed ones in the training set, is rather difficult. To tackle this issue, we propose building latent and context-based feature representations that can bring closer the input space (online/digital behaviour) across different users. In Chapter 9, we present our first steps towards generating context-based representations, demonstrating their ability to capture similarities across diverse user behaviour. Incorporating such latent user representations for the mental health assessment task is a challenging direction for future work in the domain.

10.2 Directions for Future Research

There are a few directions towards which future work can focus. In this final section, we outline some of the major such directions, based on the tasks that were tackled in this Thesis.

10.2.1 Document-level Analysis

We outline three major directions towards which future work on opinion mining in the Greek language can focus:

- Incorporate our resources for longitudinal macro-level tasks in the Greek language. Given the high accuracy we achieved in the across-domain sentiment analysis task using our resources, even with simplistic models, we encourage future work to use our findings in order to monitor the sentiment of the Greek “Twittershpere” over time and mine the resulting time series for further macro-level analysis purposes.
- Develop state-of-the-art models for the task of sentiment analysis and test the accuracy of our resources in other tasks, such as stance detection and target-specific sentiment analysis [254].
- Generate task-specific [231] word representations or more context-based representations [191], and compare the accuracy in various tasks, against our resources.

Our vision is to encourage future research in NLP in Greek and other under-resourced languages, aiming to bridge the gap between the latter and the most widely used languages.

10.2.2 Macro-level Monitoring Using Social Media

We propose the following directions for future work in the domain of nowcasting macro-level political indicators:

- Apply better feature engineering through state-of-the-art sentiment analysis methods and incorporate user network representation approaches. Since the effectiveness of the models that predict electoral outcomes can only be effectively assessed based on the election results, testing the time-series models in multiple electoral cases is also essential for providing supportive evidence on their appropriateness for the task.

- Instead of building election-specific or party-specific approaches, we can instead try to train a model based on the election results of several electoral races and test it in a held-out one, in a leave-one-election-out validation setting. This will ensure the ability of such models to generalise in new cases, without the need to rely in the (often, noisy) opinion polls.

Finally, future work can try to use such time-series models in different macro-level tasks, either in isolation or in a multi-task setting.

10.2.3 Micro-level Monitoring Using Social Media

Within our micro-level voting intention detection task, we identify the following directions for future research:

- Incorporating further external information, such as image or multimedia content, can provide further boost to MCKL. We have demonstrated that MCKL is able to cope with noisy features, thus the incorporation of further information in our modelling is safer than in feature aggregate baselines.
- While MCKL models the textual component of the task in a fine-grained granularity (message-level), the same does not hold for the network modelling part of the task. Building fine-grained network representations of the users in a temporal fashion under a real-world setting can produce a more appropriate model, especially in times of crisis, when events in the real-world occur rapidly and user opinion formation changes rather dynamically.
- Working on longer-lasting electoral races will help in providing better insights on the appropriateness of our method. We hypothesise that MCKL will achieve better results in longer-lasting cases, compared to feature aggregate models, since it will effectively capture the temporal component of the task, which proved to be of significant interest.

Finally, there is recent evidence that linguistic and network user information can

be used alongside to offer improvement in model performance in other micro-level tasks, too [5]. A potentially fruitful area of future research is that of multi-source and multi-modal fusion of such temporally sensitive information for various other micro-level tasks.

10.2.4 Micro-level Monitoring Using Heterogeneous Sources

In the previous parts of this Thesis, we have proposed several directions for future work, aiming to bring in advances to our already well-performing models. In the domain of assessing mental health using heterogeneous data sources, we have demonstrated that current state-of-the-art fail to provide appropriate real-world solutions to the task. Despite the fact that there are several advances possible from a methodological point of view, we encourage future work to focus on building and evaluating models *under a real-world setting*. In particular, we propose to build transfer learning approaches on latent user representations over time, aiming to achieve positive results in an unbiased *leave-one-user-out* setting. To this end, we have presented in Chapter 9 our first steps towards generating such latent representations using logs from the smart devices of different users, demonstrating that they can be effectively represent the behaviour of a user in a low-dimensional vector, with minimal training. Building on these representations and incorporating them into the mental health assessment task is a promising direction for future work within the domain. Models that incorporate such representations under a real-world setting could revolutionise traditional processes on mental health assessment, while offering the researchers with clear insights on the factors that affect one’s mental health.

Bibliography

- [1] P. Agathangelou, I. Katakis, F. Kokkoras, and K. Ntonas. Mining Domain-Specific Dictionaries of Opinion Words. In *International Conference on Web Information Systems Engineering*, pages 47–62. Springer, 2014.
- [2] L. M. Aiello, G. Petkos, C. Martin, D. Corney, S. Papadopoulos, R. Skraba, A. Göker, I. Kompatsiaris, and A. Jaimes. Sensing Trending Topics in Twitter. *IEEE Transactions on Multimedia*, 15(6):1268–1282, 2013.
- [3] F. Al Zamal, W. Liu, and D. Ruths. Homophily and Latent Attribute Inference: Inferring Latent Attributes of Twitter Users from Neighbors. *ICWSM*, 270:2012, 2012.
- [4] J. H. Aldrich, R. K. Gibson, M. Cantijoch, and T. Konitzer. Getting Out the Vote in the Social mMedia Era: Are Digital Tools Changing the Extent, Nature and Impact of Party Contacting in Elections? *Party Politics*, 22(2):165–178, 2016.
- [5] N. Aletras and B. P. Chamberlain. Predicting Twitter User Socioeconomic Attributes with Network and Language Information. *arXiv preprint arXiv:1804.04095*, 2018.
- [6] J. Alvarez-Lozano, V. Osmani, O. Mayora, M. Frost, J. Bardram, M. Faurholt-Jepsen, and L. V. Kessing. Tell Me Your Apps and I Will Tell You Your Mood: Correlation of Apps Usage with Bipolar Disorder State. In *Proceedings of the 7th International Conference on Pervasive Technologies Related to Assistive Environments*, page 19. ACM, 2014.
- [7] S. Amir, G. Coppersmith, P. Carvalho, M. J. Silva, and B. C. Wallace. Quantifying Mental Health from Social Media with Neural User Embeddings. *arXiv preprint arXiv:1705.00335*, 2017.
- [8] A. Andreevskaia and S. Bergler. When Specialists and Generalists Work Together: Overcoming Domain Dependence in Sentiment Tagging. *Proceedings of ACL-08: HLT*, pages 290–298, 2008.
- [9] D. Antonakaki, D. Spiliotopoulos, C. V. Samaras, P. Pratikakis, S. Ioannidis, and P. Fragopoulou. Social Media Analysis during Political Turbulence. *PloS one*, 12(10):e0186836, 2017.

- [10] P. Arora, A. Bakliwal, and V. Varma. Hindi Subjective Lexicon Generation using WordNet Graph Traversal. *International Journal of Computational Linguistics and Applications*, 3(1):25–39, 2012.
- [11] R. Artstein and M. Poesio. Inter-coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4):555–596, 2008.
- [12] S. Asur and B. A. Huberman. Predicting the Future with Social Media. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 492–499. IEEE Computer Society, 2010.
- [13] P. D. Azar and A. W. Lo. The Wisdom of Twitter Crowds: Predicting Stock Market Reactions to FOMC Meetings via Twitter Feeds. *Journal of Portfolio Management*, 42(5):123, 2016.
- [14] S. Bagroy, P. Kumaraguru, and M. De Choudhury. A Social Media Based Index of Mental Well-Being in College Campuses. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 1634–1646. ACM, 2017.
- [15] S. Balani and M. De Choudhury. Detecting and Characterizing Mental Health Related Self-Disclosure in Social Media. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pages 1373–1378. ACM, 2015.
- [16] L. Barbosa and J. Feng. Robust Sentiment Detection on Twitter from Biased and Noisy Data. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 36–44. Association for Computational Linguistics, 2010.
- [17] C. Baziotis, N. Pelekis, and C. Doukeridis. DataStories at SemEval-2017 Task 4: Deep LSTM with Attention for Message-level and Topic-based Sentiment Analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754, 2017.
- [18] A. Benton, M. Mitchell, and D. Hovy. Multitask Learning for Mental Health Conditions with Limited Social Media Data. In *Proceedings of the 15th Conference of the EACL*, volume 1, pages 152–162, 2017.
- [19] A. Bermingham and A. Smeaton. On Using Twitter to Monitor Political Sentiment and Predict Election Results. In *Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP 2011)*, pages 2–10, 2011.

- [20] A. Bermingham and A. F. Smeaton. Classifying Sentiment in Microblogs: Is Brevity an Advantage? In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1833–1836. ACM, 2010.
- [21] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.
- [22] J. Blitzer, M. Dredze, and F. Pereira. Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447, 2007.
- [23] A. Bogomolov, B. Lepri, M. Ferron, F. Pianesi, and A. S. Pentland. Daily Stress Recognition from Mobile Phone Data, Weather Conditions and Individual Traits. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 477–486. ACM, 2014.
- [24] A. Bogomolov, B. Lepri, M. Ferron, F. Pianesi, and A. S. Pentland. Pervasive Stress Recognition for Sustainable Living. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on*, pages 345–350. IEEE, 2014.
- [25] A. Bogomolov, B. Lepri, and F. Pianesi. Happiness Recognition from Mobile Phone Data. In *Social Computing (SocialCom), 2013 International Conference on*, pages 790–795. IEEE, 2013.
- [26] J. Bollen, H. Mao, and X. Zeng. Twitter Mood Predicts the Stock Market. *Journal of Computational Science*, 2(1):1–8, 2011.
- [27] J. Borge-Holthoefer, W. Magdy, K. Darwish, and I. Weber. Content and Network Dynamics behind Egyptian Political Polarization on Twitter. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 700–711. ACM, 2015.
- [28] A. Bosco and S. Verney. Electoral Epidemic: The Political Cost of Economic Crisis in Southern Europe, 2010–11. *South European Society and Politics*, 17(2):129–154, 2012.
- [29] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A Training Algorithm for Optimal Margin Classifiers. In *Proceedings of the fifth Annual Workshop on Computational Learning Theory*, pages 144–152. ACM, 1992.
- [30] A. Boutet, H. Kim, E. Yoneki, et al. What’s in Your Tweets? I Know Who You Supported in the UK 2010 General Election. *ICWSM*, 12:411–414, 2012.

- [31] M. M. Bradley and P. J. Lang. Affective Norms for English Words (ANEW): Instruction Manual and Affective Ratings. Technical report, Technical report C-1, the center for research in psychophysiology, University of Florida, 1999.
- [32] L. Breiman. Random Forests. *Machine learning*, 45(1):5–32, 2001.
- [33] C. Brew. Classifying ReachOut Posts with a Radial Basis Function SVM . In *CLPsych@ HLT-NAACL*, pages 138–142, 2016.
- [34] P. Burnap, R. Gibson, L. Sloan, R. Southern, and M. Williams. 140 Characters to Victory?: Using Twitter to Predict the UK 2015 General Election. *Electoral Studies*, 41:230–233, 2016.
- [35] M. P. Cameron, P. Barrett, and B. Stewardson. Can Social Media Predict Election Results? Evidence from New Zealand. *Journal of Political Marketing*, 15(4):416–432, 2016.
- [36] L. Canzian and M. Musolesi. Trajectories of Depression: Unobtrusive Monitoring of Depressive States by Means of Smartphone Mobility Traces Analysis. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pages 1293–1304. ACM, 2015.
- [37] A. Ceron, L. Curini, S. M. Iacus, and G. Porro. Every Tweet Counts? How Sentiment Analysis of Social Media can Improve Our Knowledge of Citizens’ Political Preferences with an Application to Italy and France. *New Media & Society*, 16(2):340–358, 2014.
- [38] W. Chen and K.-H. Lee. Sharing, Liking, Commenting, and Distressed? The Pathway between Facebook Interaction and Psychological Distress. *Cyberpsychology, Behavior, and Social Networking*, 16(10):728–734, 2013.
- [39] J. E. Chung and E. Mustafaraj. Can Collective Sentiment Expressed on Twitter Predict Political Elections? In *AAAI*, volume 11, pages 1770–1771, 2011.
- [40] K. W. Church and P. Hanks. Word Association Norms, Mutual Information, and Lexicography. *Computational linguistics*, 16(1):22–29, 1990.
- [41] M. Cliche. BB.twtr at SemEval-2017 Task 4: Twitter Sentiment Analysis with CNNs and LSTMs. pages 573–580, 2017.
- [42] D. L. Cogburn and F. K. Espinoza-Vasquez. From Networked Nominee to Networked Nation: Examining the Impact of Web 2.0 and Social Media on Political Participation and Civic Engagement in the 2008 Obama Campaign. *Journal of Political Marketing*, 10(1-2):189–213, 2011.

- [43] A. Cohan, S. Young, and N. Goharian. Triaging Mental Health Forum Posts. In *CLPsych@ HLT-NAACL*, pages 143–147, 2016.
- [44] R. Cohen and D. Ruths. Classifying Political Orientation on Twitter: It’s not Easy! In *ICWSM*, 2013.
- [45] E. Colleoni, A. Rozza, and A. Arvidsson. Echo Chamber or Public Sphere? Predicting Political Orientation and Measuring Political Homophily in Twitter using Big Data. *Journal of Communication*, 64(2):317–332, 2014.
- [46] M. Collins and N. Duffy. Convolution Kernels for Natural Language. In *NIPS*, pages 625–632, 2002.
- [47] M. Conover, J. Ratkiewicz, M. R. Francisco, B. Gonçalves, F. Menczer, and A. Flammini. Political Polarization on Twitter. In *ICWSM*, volume 133, pages 89–96, 2011.
- [48] M. Constant and A. Yannacopoulou. Le Dictionnaire électronique du Grec Moderne: Conception et développement d’outils pour son Enrichissement et sa Validation. In *Studies in Greek Linguistics, Proceedings of the 23rd annual meeting of the Department of Linguistics (2002)*, volume 2, pages 783–791. Faculty of Philosophy, Aristotle University of Thessaloniki, 2003.
- [49] G. Coppersmith, M. Dredze, C. Harman, and K. Hollingshead. From ADHD to SAD: Analyzing the Language of Mental Health on Twitter through Self-Reported Diagnoses. In *CLPsych@ HLT-NAACL*, pages 1–10, 2015.
- [50] G. Coppersmith, M. Dredze, C. Harman, K. Hollingshead, and M. Mitchell. CLPsych 2015 Shared Task: Depression and PTSD on Twitter. In *CLPsych@ HLT-NAACL*, pages 31–39, 2015.
- [51] G. Coppersmith, K. Ngo, R. Leary, and A. Wood. Exploratory Analysis of Social Media Prior to a Suicide Attempt. In *CLPsych@ HLT-NAACL*, pages 106–117, 2016.
- [52] C. Cortes and V. Vapnik. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995.
- [53] D. R. Cox. The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 215–242, 1958.
- [54] J. Crawford and J. Henry. The Positive and Negative Affect Schedule (PANAS): Construct validity, measurement properties and normative data in a large non-clinical sample. *British Journal of Clinical Psychology*, 43, 2004.

- [55] A. Culotta. Towards Detecting Influenza Epidemics by Analyzing Twitter Messages. In *Proceedings of the first Workshop on Social Media Analytics*, pages 115–122. ACM, 2010.
- [56] A. Das and S. Bandyopadhyay. SentiWordNet for Indian Languages. *Asian Federation for Natural Language Processing, China*, pages 56–63, 2010.
- [57] M. De Choudhury, S. Counts, and E. Horvitz. Social Media as a Measurement Tool of Depression in Populations. In *Proceedings of the 5th Annual ACM Web Science Conference*, pages 47–56. ACM, 2013.
- [58] M. De Choudhury, M. Gamon, S. Counts, and E. Horvitz. Predicting Depression via Social Media. In *ICWSM*, page 2, 2013.
- [59] M. De Choudhury, E. Kiciman, M. Dredze, G. Coppersmith, and M. Kumar. Discovering Shifts to Suicidal Ideation from Mental Health Content in Social Media. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 2098–2110. ACM, 2016.
- [60] M. Del Tredici and R. Fernández. Semantic Variation in Online Communities of Practice. In *IWCS*, 2017.
- [61] O. DeMasi, K. Kording, and B. Recht. Meaningless Comparisons Lead to False Optimism in Medical Machine Learning. *PLoS One*, 12(9):e0184604, 2017.
- [62] L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. W. S. Hoi, and A. Zubiaga. SemEval-2017 Task 8: RumourEval: Determining Rumour Veracity and Support for Rumours. In *SemEval*, pages 69–76, 2017.
- [63] J. DiGrazia, K. McKelvey, J. Bollen, and F. Rojas. More Tweets, More Votes: Social Media as a Quantitative Indicator of Political Behavior. *PloS one*, 8(11):e79449, 2013.
- [64] X. Ding, B. Liu, and P. S. Yu. A Holistic Lexicon-based Approach to Opinion Mining. In *Proceedings of the 2008 international conference on web search and data mining*, pages 231–240. ACM, 2008.
- [65] P. S. Dodds, K. D. Harris, I. M. Kloumann, C. A. Bliss, and C. M. Danforth. Temporal Patterns of Happiness and Information in a Global Social Network: Hedonometrics and Twitter. *PloS one*, 6(12):e26752, 2011.
- [66] H. Drucker, C. J. Burges, L. Kaufman, A. J. Smola, and V. Vapnik. Support Vector Regression Machines. In *Advances in Neural Information Processing Systems*, pages 155–161, 1997.

- [67] J. C. Eichstaedt, H. A. Schwartz, M. L. Kern, G. Park, D. R. Labarthe, R. M. Merchant, S. Jha, M. Agrawal, L. A. Dziurzynski, M. Sap, et al. Psychological Language on Twitter Predicts County-Level Heart Disease Mortality. *Psychological science*, 26(2):159–169, 2015.
- [68] P. Ekman. An Argument for Basic Emotions. *Cognition & emotion*, 6(3-4):169–200, 1992.
- [69] S. R. El-Beltagy, A. B. Soliman, et al. NileTMRG at SemEval-2017 Task 4: Arabic Sentiment Analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 790–795, 2017.
- [70] C. Esteban, D. Schmidt, D. Krompaß, and V. Tresp. Predicting Sequences of Clinical Events by Using a Personalized Temporal Latent Embedding Model. In *Healthcare Informatics (ICHI), 2015 International Conference on*, pages 130–139. IEEE, 2015.
- [71] A. Esuli and F. Sebastiani. SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining. In *Proceedings of LREC*, volume 6, pages 417–422. Citeseer, 2006.
- [72] A. Fang, I. Ounis, P. Habel, C. Macdonald, and N. Limsopatham. Topic-Centric Classification of Twitter User’s Political Orientation. In *SIGIR*, pages 791–794, 2015.
- [73] A. A. Farhan, C. Yue, R. Morillo, S. Ware, J. Lu, J. Bi, J. Kamath, A. Russell, A. Bamis, and B. Wang. Behavior vs. Introspection: Refining Prediction of Clinical Depression via Smartphone Sensing Data. In *Wireless Health*, pages 30–37, 2016.
- [74] L. Floridi and M. Taddeo. What is data ethics? *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2083), 2016.
- [75] M. J. Forgeard, E. Jayawickreme, M. L. Kern, and M. E. Seligman. Doing the right thing: Measuring wellbeing for public policy. *International journal of wellbeing*, 1(1), 2011.
- [76] G. W. Furnas, S. Deerwester, S. T. Dumais, T. K. Landauer, R. A. Harshman, L. A. Streeter, and K. E. Lochbaum. Information Retrieval using a Singular Value Decomposition Model of Latent Semantic Structure. In *Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 465–480. ACM, 1988.

- [77] K. Garimella, G. De Francisci Morales, A. Gionis, and M. Mathioudakis. Political Discourse on Social Media: Echo Chambers, Gatekeepers, and the Price of Bipartisanship. In *WWW*, pages 913–922, 2018.
- [78] D. Gayo-Avello. I Wanted to Predict Elections with Twitter and all I Got was this Lousy Paper – A Balanced Survey on Election Prediction using Twitter Data. *arXiv preprint arXiv:1204.6441*, 2012.
- [79] D. Gayo-Avello. A Meta-Analysis of State-of-the-Art Electoral Prediction from Twitter Data. *Social Science Computer Review*, 31(6):649–679, 2013.
- [80] D. Gayo-Avello. Politics and Social Meedia. *Unpublished manuscript*, 2016.
- [81] D. Gayo Avello, P. T. Metaxas, and E. Mustafaraj. Limits of Electoral Predictions Using Twitter. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*. Association for the Advancement of Artificial Intelligence, 2011.
- [82] M. S. Gerber. Predicting Crime Using Twitter and Kernel Density Estimation. *Decision Support Systems*, 61:115–125, 2014.
- [83] K. Gimpel, N. Schneider, B. O’Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics, 2011.
- [84] A. Go, R. Bhayani, and L. Huang. Twitter Sentiment Classification using Distant Supervision. *CS224N Project Report, Stanford*, 1:12, 2009.
- [85] Y. Goldberg and O. Levy. word2vec Explained: Deriving Mikolov et al.’s Negative-Sampling Word-Embedding Method. *arXiv preprint arXiv:1402.3722*, 2014.
- [86] P. Gonçalves, M. Araújo, F. Benevenuto, and M. Cha. Comparing and Combining Sentiment Analysis Methods. In *Proceedings of the first ACM conference on Online social networks*, pages 27–38. ACM, 2013.
- [87] J.-A. González, F. Pla, and L.-F. Hurtado. ELiRF-UPV at SemEval-2017 Task 4: Sentiment Analysis using Deep Learning. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 723–727, 2017.

- [88] R. González-Ibáñez, S. Muresan, and N. Wacholder. Identifying Sarcasm in Twitter: a Closer Look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 581–586. Association for Computational Linguistics, 2011.
- [89] R. Gupta, N. Malandrakis, B. Xiao, T. Guha, M. Van Segbroeck, M. Black, A. Potamianos, and S. Narayanan. Multimodal Prediction of Affective Dimensions and Depression in Human-Computer Interactions. In *Proceedings of the 4th International Workshop on Audio/Visual Emotion Challenge*, pages 33–40. ACM, 2014.
- [90] M. Gutmann and A. Hyvärinen. Noise-Contrastive Estimation: A New Estimation Principle for Unnormalized Statistical Models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304, 2010.
- [91] G. A. Harman, C. T. Coppersmith, and M. H. Dredze. Measuring Post Traumatic Stress Disorder in Twitter. In *ICWSM*, 2014.
- [92] D. Haussler. Convolution Kernels on Discrete Structures. Technical report, Department of Computer Science, University of California at Santa Cruz, 1999.
- [93] B. Heredia, J. D. Prusa, and T. M. Khoshgoftaar. The impact of malicious accounts on political tweet sentiment. In *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*, pages 197–202. IEEE, 2018.
- [94] H. Herrman, S. Saxena, R. Moodie, et al. *Promoting mental health: concepts, emerging evidence, practice: a report of the World Health Organization, Department of Mental Health and Substance Abuse in collaboration with the Victorian Health Promotion Foundation and the University of Melbourne*. World Health Organization, 2005.
- [95] T. K. Ho. Random Decision Forests. In *Document Analysis and Recognition, 1995., Proceedings of the third International Conference on*, volume 1, pages 278–282. IEEE, 1995.
- [96] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural computation*, 9(8):1735–1780, 1997.
- [97] T. Hofmann, B. Schölkopf, and A. J. Smola. Kernel Methods in Machine Learning. *The Annals of Statistics*, pages 1171–1220, 2008.

- [98] K. Hornik, M. Stinchcombe, and H. White. Multilayer Feedforward Networks are Universal Approximators. *Neural networks*, 2(5):359–366, 1989.
- [99] P. N. Howard. Deep Democracy, Thin Citizenship: The Impact of Digital Media in Political Campaign Strategy. *The Annals of the American Academy of Political and Social Science*, 597(1):153–170, 2005.
- [100] M. Hu and B. Liu. Mining and Summarizing Customer Reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 168–177. ACM, 2004.
- [101] Y. Huang, H. Xiong, K. Leach, Y. Zhang, P. Chow, K. Fua, B. A. Teichman, and L. E. Barnes. Assessing social anxiety using gps trajectories and point-of-interest data. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 898–903. ACM, 2016.
- [102] M. Jabreel and A. Moreno. SiTAKA at SemEval-2017 Task 4: Sentiment Analysis in Twitter Based on a Rich Set of Features. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 694–699, 2017.
- [103] N. Jaques, S. Taylor, A. Azaria, A. Ghandeharioun, A. Sano, and R. Picard. Predicting Students’ Happiness from Physiology, Phone, Mobility, and Behavioral Data. In *Affective computing and intelligent interaction (ACII), 2015 international conference on*, pages 222–228. IEEE, 2015.
- [104] N. Jaques, S. Taylor, A. Sano, and R. Picard. Multi-Task, Multi-Kernel Learning for Estimating Individual Wellbeing. In *NIPS Workshop on Multimodal Machine Learning*, volume 898, 2015.
- [105] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao. Target-dependent Twitter Sentiment Classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 151–160. Association for Computational Linguistics, 2011.
- [106] V. Jijkoun, M. de Rijke, and W. Weerkamp. Generating Focused Topic-Specific Sentiment Lexicons. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 585–594. Association for Computational Linguistics, 2010.
- [107] V. Jijkoun and K. Hofmann. Generating a Non-English Subjectivity Lexicon: Relations That Matter. In *Proceedings of the 12th Conference of*

- the European Chapter of the Association for Computational Linguistics*, pages 398–405. Association for Computational Linguistics, 2009.
- [108] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *ECML*, pages 137–142, 1998.
- [109] C. Jørgensen. Public debate—an act of hostility? *Argumentation*, 12(4):431–443, 1998.
- [110] A. Jungherr. Twitter use in election campaigns: A systematic literature review. *Journal of information technology & politics*, 13(1):72–91, 2016.
- [111] A. Jungherr, P. Jürgens, and H. Schoen. Why the Pirate Party won the German Election of 2009 or the Trouble with Predictions: A response to Tumasjan, A., Sprenger, T.O., Sander, P.G., & welpe, I.M. “Predicting elections with Twitter: What 140 characters reveal about political sentiment”. *Social science computer review*, 30(2):229–234, 2012.
- [112] G. Kalamatianos, D. Mallis, S. Symeonidis, and A. Arampatzis. Sentiment Analysis of Greek Tweets and Hashtags using a Sentiment Lexicon. In *Proceedings of the 19th Panhellenic Conference on Informatics*, pages 63–68. ACM, 2015.
- [113] E. Kalampokis, A. Karamanou, E. Tambouris, and K. A. Tarabanis. On Predicting Election Results using Twitter and Linked Open Data: The Case of the UK 2010 Election. *J. UCS*, 23(3):280–303, 2017.
- [114] N. Kalchbrenner, E. Grefenstette, and P. Blunsom. A Convolutional Neural Network for Modelling Sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 655–665, 2014.
- [115] A. M. Kaplan and M. Haenlein. Users of the world, unite! the challenges and opportunities of social media. *Business horizons*, 53(1):59–68, 2010.
- [116] H. Kennedy and G. Moss. Known or knowing publics? social media data mining and the question of public agency. *Big Data & Society*, 2(2):2053951715611145, 2015.
- [117] A. Z. Khan, M. Atique, and V. Thakare. Combining Lexicon-based and Learning-based Methods for Twitter Sentiment Analysis. *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCE)*, page 89, 2015.

- [118] J. Kim, F. Rousseau, and M. Vazirgiannis. Convolutional sentence kernel from word embeddings for short text categorization. In *EMNLP*, pages 775–780, 2015.
- [119] Y. Kim. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, 2014.
- [120] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [121] S. Kiritchenko, X. Zhu, and S. M. Mohammad. Sentiment analysis of short informal texts. 50:723–762, 2014.
- [122] K. Kroenke, T. W. Strine, R. L. Spitzer, J. B. Williams, J. T. Berry, and A. H. Mokdad. The phq-8 as a measure of current depression in the general population. *Journal of affective disorders*, 114(1):163–173, 2009.
- [123] T. Kyriacopoulou. Analyse automatique des textes écrits: le cas du grec moderne, 2004.
- [124] V. Lampos, T. De Bie, and N. Cristianini. Flu Detector-Tracking Epidemics on Twitter. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 599–602. Springer, 2010.
- [125] V. Lampos, T. Lansdall-Welfare, R. Araya, and N. Cristianini. Analysing mood patterns in the united kingdom through twitter content. *arXiv preprint arXiv:1304.5507*, 2013.
- [126] V. Lampos, D. Preoțiuc-Pietro, and T. Cohn. A user-centric model of voting intention from social media. In *ACL*, volume 1, pages 993–1003, 2013.
- [127] V. Lampos, B. Zou, and I. J. Cox. Enhancing Feature Selection Using Word Embeddings: The Case of Flu Surveillance. In *Proceedings of the 26th International Conference on World Wide Web*, pages 695–704. International World Wide Web Conferences Steering Committee, 2017.
- [128] T. Lansdall-Welfare, V. Lampos, and N. Cristianini. Effects of the recession on public mood in the uk. In *Proceedings of the 21st International Conference on World Wide Web*, pages 1221–1226. ACM, 2012.
- [129] R. J. Larsen, E. Diener, and R. A. Emmons. An evaluation of subjective well-being measures. *Social indicators research*, 17(1):1–17, 1985.

- [130] N. Lathia, D. Quercia, and J. Crowcroft. The hidden image of the city: sensing community well-being from urban mobility. In *International conference on pervasive computing*, pages 91–98. Springer, 2012.
- [131] A. Lazaridou, N. T. Pham, and M. Baroni. Combining language and vision with a multimodal skip-gram model. *arXiv preprint arXiv:1501.02598*, 2015.
- [132] Q. Le and T. Mikolov. Distributed Representations of Sentences and Documents. In *International Conference on Machine Learning*, pages 1188–1196, 2014.
- [133] S. Le Cessie and J. C. Van Houwelingen. Ridge estimators in logistic regression. *Applied statistics*, pages 191–201, 1992.
- [134] R. LiKamWa, Y. Liu, N. D. Lane, and L. Zhong. Moodscope: Building a mood sensor from smartphone usage patterns. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pages 389–402. ACM, 2013.
- [135] H. Lin, W. Tov, and L. Qiu. Emotional disclosure on social networking sites: the role of network structure and psychological needs. *Computers in Human Behavior*, 41:342–350, 2014.
- [136] L. Liu, L. Tang, W. Dong, S. Yao, and W. Zhou. An Overview of Topic Modeling and its Current Applications in Bioinformatics. *SpringerPlus*, 5(1):1608, 2016.
- [137] P. Liu, W. Tov, M. Kosinski, D. J. Stillwell, and L. Qiu. Do facebook status updates reflect subjective well-being? *Cyberpsychology, Behavior, and Social Networking*, 18(7):373–379, 2015.
- [138] A. Llorente, M. Garcia-Herranz, M. Cebrian, and E. Moro. Social Media Fingerprints of Unemployment. *PloS one*, 10(5):e0128692, 2015.
- [139] W.-Y. Loh. Fifty Years of Classification and Regression Trees. *International Statistical Review*, 82(3):329–348, 2014.
- [140] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell. The jigsaw continuous sensing engine for mobile phone applications. In *Proceedings of the 8th ACM conference on embedded networked sensor systems*, pages 71–84. ACM, 2010.
- [141] M. Lukasik and T. Cohn. Convolution Kernels for Discriminative Learning from Streaming Text. In *AAAI*, pages 2757–2763, 2016.

- [142] J. Ma, W. Gao, Z. Wei, Y. Lu, and K.-F. Wong. Detect Rumors using Time Series of Social Context Information on Microblogging Websites. In *CIKM*, pages 1751–1754, 2015.
- [143] Y. Ma, H. Peng, and E. Cambria. Targeted Aspect-based Sentiment Analysis via Embedding Commonsense Knowledge into an Attentive LSTM. In *Proceedings of AAAI*, 2018.
- [144] Y. Ma, B. Xu, Y. Bai, G. Sun, and R. Zhu. Daily mood assessment based on mobile phone sensing. In *Wearable and implantable body sensor networks (BSN), 2012 ninth international conference on*, pages 142–147. IEEE, 2012.
- [145] L. v. d. Maaten and G. Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.
- [146] S. Mac Kim, Y. Wang, S. Wan, and C. Paris. Data61-CSIRO Systems at the CLPsych 2016 Shared Task. In *CLPsych@ HLT-NAACL*, pages 128–132, 2016.
- [147] S. Malmasi, M. Zampieri, and M. Dras. Predicting post severity in mental health forums. *order*, 2:8, 2016.
- [148] A. Markham and E. Buchanan. Ethical decision-making and internet research: Version 2.0. recommendations from the AoIR ethics working committee. Available online: aoir.org/reports/ethics2.pdf, 2012.
- [149] A. Mehrotra, R. Hendley, and M. Musolesi. Towards multi-modal anticipatory monitoring of depressive states through the analysis of human-smartphone interaction. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, pages 1132–1138. ACM, 2016.
- [150] P. T. Metaxas, E. Mustafaraj, and D. Gayo-Avello. How (not) to predict elections. In *Privacy, Security, Risk and Trust and IEEE Third International Conference on Social Computing*, pages 165–171, 2011.
- [151] A. Michailidou. Twitter, Public Engagement and the Eurocrisis: More than an Echo Chamber? In *Social Media and European Politics*, pages 241–266. Springer, 2017.
- [152] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*, 2013.

- [153] T. Mikolov and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. *Advances in neural information processing systems*, 2013.
- [154] G. A. Miller. WordNet: a Lexical Database for English. *Communications of the ACM*, 38(11):39–41, 1995.
- [155] D. N. Milne, G. Pink, B. Hachey, and R. A. Calvo. CLPsych 2016 Shared Task: Triaging Content in Online Peer-Support Forums. In *CLPsych@ HLT-NAACL*, pages 118–127, 2016.
- [156] R. Miranda Filho, J. M. Almeida, and G. L. Pappa. Twitter population sample bias and its impact on predictive outcomes: a case study on elections. In *Advances in Social Networks Analysis and Mining*, pages 1254–1261, 2015.
- [157] M. Mitchell, K. Hollingshead, and G. Coppersmith. Quantifying the Language of Schizophrenia in Social Media. In *CLPsych@ HLT-NAACL*, pages 11–20, 2015.
- [158] S. Mohammad. #emotional tweets. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 246–255, Montréal, Canada, 7-8 June 2012. Association for Computational Linguistics.
- [159] S. Mohammad, C. Dunne, and B. Dorr. Generating High-Coverage Semantic Orientation Lexicons from Overtly Marked Words and a Thesaurus. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 599–608. Association for Computational Linguistics, 2009.
- [160] S. Mohammad, S. Kiritchenko, and X. Zhu. NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. In *SemEval*, volume 2, pages 321–327, 2013.
- [161] S. M. Mohammad and F. Bravo-Marquez. Emotion intensities in tweets. In *Proceedings of the sixth joint conference on lexical and computational semantics (*Sem)*, Vancouver, Canada, 2017.
- [162] S. M. Mohammad, M. Salameh, and S. Kiritchenko. How Translation Alters Sentiment. *J. Artif. Intell. Res.(JAIR)*, 55:95–130, 2016.

-
- [163] S. M. Mohammad and P. D. Turney. Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*, pages 26–34. Association for Computational Linguistics, 2010.
- [164] S. M. Mohammad and P. D. Turney. Crowdsourcing a Word–Emotion Association Lexicon. *Computational Intelligence*, 29(3):436–465, 2013.
- [165] A. Moore and P. Rayson. Bringing Replication and Reproduction Together with Generalisability in NLP: Three Reproduction Studies for Target Dependent Sentiment Analysis. *arXiv preprint arXiv:1806.05219*, 2018.
- [166] A. Morales, J. Borondo, J. C. Losada, and R. M. Benito. Measuring Political Polarization: Twitter Shows the Two Sides of Venezuela. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 25(3):033114, 2015.
- [167] M. A. Moreno, L. A. Jelenchick, K. G. Egan, E. Cox, H. Young, K. E. Gannon, and T. Becker. Feeling bad on facebook: depression disclosures by college students on a social networking site. *Depression and anxiety*, 28(6):447–455, 2011.
- [168] F. Morin and Y. Bengio. Hierarchical Probabilistic Neural Network Language Model. In *Aistats*, volume 5, pages 246–252. Citeseer, 2005.
- [169] F. Morstatter, J. Pfeffer, H. Liu, and K. M. Carley. Is the Sample Good Enough? Comparing data from Twitter’s streaming api with twitter’s firehose. In *ICWSM*, 2013.
- [170] S. T. Moturu, I. Khayal, N. Aharony, W. Pan, and A. Pentland. Using social sensing to understand the links between sleep, mood, and sociability. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third International Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 208–214. IEEE, 2011.
- [171] R. Navigli and S. P. Ponzetto. BabelNet: The Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network. *Artificial Intelligence*, 193:217–250, 2012.
- [172] F. Å. Nielsen. A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903*, 2011.
- [173] B. O’Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *ICWSM*, volume 11, pages 1–2, 2010.

- [174] OECD. How's life? 2013: Measuring well-being. 2013.
- [175] J. Olesen, A. Gustavsson, M. Svensson, H.-U. Wittchen, and B. Jönsson. The economic cost of brain disorders in europe. *European journal of neurology*, 19(1):155–162, 2012.
- [176] N. Oliveira, P. Cortez, and N. Areal. The Impact of Microblogging Data for Stock Market Prediction: Using Twitter to Predict Returns, Volatility, Trading Volume and Survey Sentiment Indices. *Expert Systems with Applications*, 73:125–144, 2017.
- [177] V. Osmani, A. Maxhuni, A. Grünerbl, P. Lukowicz, C. Haring, and O. Mayora. Monitoring activity of patients with bipolar disorder using smart phones. In *Proceedings of International Conference on Advances in Mobile Computing & Multimedia*, page 85. ACM, 2013.
- [178] A. Pak and P. Paroubek. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In *Proceedings of the 7th Conference on International Language Resources and Evaluation (LREC'10)*, 2010.
- [179] E. Palogiannidi, E. Iosif, P. Koutsakis, and A. Potamianos. Affective Lexicon Creation for the Greek Language. *submitted to* SEM*, 2015.
- [180] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.
- [181] B. Pang, L. Lee, et al. Opinion Mining and Sentiment Analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135, 2008.
- [182] M. J. Paul and M. Dredze. You Are What You Tweet: Analyzing Twitter for Public Health. *ICWSM*, 20:265–272, 2011.
- [183] U. Pavalanathan and J. Eisenstein. Confounds and consequences in geo-tagged twitter data. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2138–2148, 2015.
- [184] T. Pedersen. Screening Twitter Users for Depression and PTSD with Lexical Decision Lists. In *CLPsych@ HLT-NAACL*, pages 46–53, 2015.
- [185] V. Pejovic, N. Lathia, C. Mascolo, and M. Musolesi. Mobile-based experience sampling for behaviour research. In *Emotions and Personality in Personalized Services*, pages 141–161. Springer, 2016.
- [186] M. Pennacchiotti and A.-M. Popescu. A Machine Learning Approach to Twitter User Classification. *ICWSM*, 11(1):281–288, 2011.

- [187] J. W. Pennebaker, M. E. Francis, and R. J. Booth. Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, 71(2001):2001, 2001.
- [188] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.
- [189] V. Perez-Rosas, C. Banea, and R. Mihalcea. Learning Sentiment Lexicons in Spanish. In *LREC*, volume 12, page 73, 2012.
- [190] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online Learning of Social Representations. In *SIGKDD*, pages 701–710, 2014.
- [191] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 2227–2237, 2018.
- [192] J. Platt. Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. 1998.
- [193] P. M. Polgreen, Y. Chen, D. M. Pennock, F. D. Nelson, and R. A. Weinstein. Using Internet Searches for Influenza Surveillance. *Clinical Infectious Diseases*, 47(11):1443–1448, 2008.
- [194] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, A.-S. Mohammad, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, et al. SemEval-2016 Task 5: Aspect based Sentiment Analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, 2016.
- [195] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos. SemEval-2015 Task 12: Aspect based Sentiment Analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, 2015.
- [196] S. Poria, H. Peng, A. Hussain, N. Howard, and E. Cambria. Ensemble Application of Convolutional Neural Networks and Multiple Kernel Learning for Multimodal Sentiment Analysis. *Neurocomputing*, 261:217–230, 2017.
- [197] M. F. Porter. An Algorithm for Suffix Stripping. *Program*, 14(3):130–137, 1980.

- [198] D. Preotiuc-Pietro, J. Eichstaedt, G. Park, M. Sap, L. Smith, V. Tobolsky, H. A. Schwartz, and L. Ungar. The role of personality, age and gender in tweeting about mental illnesses. In *NAACL HLT*, volume 2015, page 21, 2015.
- [199] D. Preotiuc-Pietro, V. Lampos, and N. Aletras. An Analysis of the User Occupational Class through Twitter Content. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1754–1764, 2015.
- [200] D. Preotiuc-Pietro, Y. Liu, D. Hopkins, and L. Ungar. Beyond Binary Labels: Political Ideology Prediction of Twitter Users. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 729–740, 2017.
- [201] D. Preotiuc-Pietro, M. Sap, H. A. Schwartz, and L. H. Ungar. Mental illness detection at the world well-being project for the clpsych 2015 shared task. In *CLPsych@ HLT-NAACL*, pages 40–45, 2015.
- [202] D. Preotiuc-Pietro, S. Volkova, V. Lampos, Y. Bachrach, and N. Aletras. Studying User Income through Language, Behaviour and Affect in Social Media. *PloS one*, 10(9):e0138717, 2015.
- [203] M. Purver and S. Battersby. Experimenting with Distant Supervision for Emotion Classification. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 482–491. Association for Computational Linguistics, 2012.
- [204] K. K. Rachuri, M. Musolesi, C. Mascolo, P. J. Rentfrow, C. Longworth, and A. Aucinas. Emotionsense: a mobile phones based adaptive platform for experimental social psychology research. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*, pages 281–290. ACM, 2010.
- [205] V. Ramanathan, K. Tang, G. Mori, and L. Fei-Fei. Learning temporal embeddings for complex video analysis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4471–4479, 2015.
- [206] G. Ranco, D. Aleksovski, G. Caldarelli, M. Grčar, and I. Mozetič. The Effects of Twitter Sentiment on Stock Price Returns. *PloS one*, 10(9):e0138441, 2015.

-
- [207] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta. Classifying Latent User Attributes in Twitter. In *Proceedings of the 2nd International Workshop on Search and Mining User-generated Contents*, pages 37–44. ACM, 2010.
- [208] J. Read. Using Emoticons to Reduce Dependency in Machine Learning Techniques for Sentiment Classification. In *Proceedings of the ACL student research workshop*, pages 43–48. Association for Computational Linguistics, 2005.
- [209] P. Resnik, W. Armstrong, L. Claudino, and T. Nguyen. The University of Maryland CLPsych 2015 Shared Task System. In *CLPsych@ HLT-NAACL*, pages 54–60, 2015.
- [210] G. Rizos, S. Papadopoulos, and Y. Kompatsiaris. Multilabel User Classification Using the Community Structure of Online Networks. *PloS one*, 12(3):e0173347, 2017.
- [211] S. Rosenthal, P. Nakov, S. Kiritchenko, S. Mohammad, A. Ritter, and V. Stoyanov. SemEval-2015 Task 10: Sentiment Analysis in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 451–463, 2015.
- [212] W. Rüdiger and G. Karyotis. Beyond the usual suspects? New participants in anti-austerity protests in Greece. *Mobilization: An International Quarterly*, 18(3):313–330, 2013.
- [213] H. Saif, Y. He, and H. Alani. Semantic Sentiment Analysis of Twitter. *The Semantic Web-ISWC 2012*, pages 508–524, 2012.
- [214] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM, 2010.
- [215] G. Salton and C. Buckley. Term-weighting Approaches in Automatic Text Retrieval. *Information Processing & Management*, 24(5):513–523, 1988.
- [216] E. T. K. Sang and J. Bos. Predicting the 2011 Dutch Senate Election Results with Twitter. In *Proceedings of the Workshop on Semantic Analysis in Social Media*, pages 53–60. Association for Computational Linguistics, 2012.
- [217] A. Sano and R. W. Picard. Stress recognition using wearable sensors and mobile phones. In *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*, pages 671–676. IEEE, 2013.

- [218] E. Schinas, S. Papadopoulos, S. Diplaris, Y. Kompatsiaris, Y. Mass, J. Herzig, and L. Boudakidis. Eventsense: Capturing the Pulse of Large-Scale Events by Mining Social Media Streams. In *Proceedings of the 17th Panhellenic Conference on Informatics*, pages 17–24. ACM, 2013.
- [219] H. A. Schwartz, J. Eichstaedt, M. L. Kern, G. Park, M. Sap, D. Stillwell, M. Kosinski, and L. Ungar. Towards assessing changes in degree of depression through facebook. In *Proceedings of the Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, pages 118–125, 2014.
- [220] H. A. Schwartz, M. Sap, M. L. Kern, J. C. Eichstaedt, A. Kapelner, M. Agrawal, E. Blanco, L. Dziurzynski, G. Park, D. Stillwell, et al. Predicting individual well-being through the language of social media. In *Pac Symp Biocomput*, volume 21, pages 516–527, 2016.
- [221] S. Servia-Rodríguez, K. K. Rachuri, C. Mascolo, P. J. Rentfrow, N. Lathia, and G. M. Sandstrom. Mobile sensing at the service of mental well-being: a large-scale longitudinal study. In *Proceedings of the 26th International Conference on World Wide Web*, pages 103–112. International World Wide Web Conferences Steering Committee, 2017.
- [222] M. Settanni and D. Marengo. Sharing feelings online: studying emotional well-being via automated text analysis of facebook posts. *Frontiers in psychology*, 6, 2015.
- [223] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos. Tensor Decomposition for Signal Processing and Machine Learning. *IEEE Transactions on Signal Processing*, 65(13):3551–3582, 2017.
- [224] M. Skoric, N. Poor, P. Achananuparp, E.-P. Lim, and J. Jiang. Tweets and Votes: A Study of the 2011 Singapore General Election. In *System Science (HICSS), 2012 45th Hawaii International Conference on*, pages 2583–2591. IEEE, 2012.
- [225] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222, 2004.
- [226] G. S. Solakidis, K. N. Vavliakis, and P. A. Mitkas. Multilingual Sentiment Analysis using Emoticons and Keywords. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014 IEEE/WIC/ACM International Joint Conferences on*, volume 2, pages 102–109. IEEE, 2014.

- [227] S. Sonnenburg, G. Rätsch, C. Schäfer, and B. Schölkopf. Large Scale Multiple Kernel Learning. *Journal of Machine Learning Research*, 7(Jul):1531–1565, 2006.
- [228] I. Stewart, Y. Pinter, and J. Eisenstein. Sí no, qué penses? catalonian independence and linguistic identity on social media. In *NAACL-HLT*, 2018.
- [229] Y. Suhara, Y. Xu, and A. Pentland. Deepmood: Forecasting depressed mood based on self-reported histories via recurrent neural networks. In *Proceedings of the 26th International Conference on World Wide Web*, pages 715–724. International World Wide Web Conferences Steering Committee, 2017.
- [230] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede. Lexicon-based Methods for Sentiment Analysis. *Computational linguistics*, 37(2):267–307, 2011.
- [231] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin. Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification. In *ACL (1)*, pages 1555–1565, 2014.
- [232] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei. LINE: Large-Scale Information Network Embedding. In *WWW*, pages 1067–1077, 2015.
- [233] J. Taylor and C. Pagliari. Mining social media data: How are research sponsors and researchers addressing the ethical challenges? *Research Ethics*, 14(2):1–39, 2018.
- [234] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Sharing Clusters among Related Groups: Hierarchical Dirichlet Processes. In *Advances in Neural Information Processing Systems*, pages 1385–1392, 2005.
- [235] R. Tennant, L. Hiller, R. Fishwick, S. Platt, S. Joseph, S. Weich, J. Parkinson, J. Secker, and S. Stewart-Brown. The Warwick-Edinburgh Mental Well-Being Scale (WEMWBS): Development and UK Validation. *Health and Quality of life Outcomes*, 5(1):63, 2007.
- [236] E. Teperoglou and E. Tsatsanis. Dealignment, De-legitimation and the Implosion of the Two-Party System in Greece: The Earthquake Election of 6 May 2012. *Journal of Elections, Public Opinion and Parties*, 24(2):222–242, 2014.
- [237] W. Tov, K. L. Ng, H. Lin, and L. Qiu. Detecting well-being via computerized content analysis of brief diary entries. *Psychological assessment*, 25(4):1069, 2013.

-
- [238] R. Townsend, A. Tsakalidis, Y. Zhou, B. Wang, M. Liakata, A. Zubiaga, A. Cristea, and R. Procter. WarwickDCS: From Phrase-based to Target-specific Sentiment Recognition. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 657–663, 2015.
- [239] Ί. Μ. Τριανταφυλλίδης. λεξικό της κοινής νεοελληνικής. Θεσσαλονίκη: Ινστιτούτο Νεοελληνικών Σπουδών του ΑΠΘ, 1998.
- [240] A. Tsakalidis, N. Aletras, A. I. Cristea, and M. Liakata. Nowcasting the Stance of Social Media Users in a Sudden Vote: The Case of the Greek Referendum. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 367–376. ACM, 2018.
- [241] A. Tsakalidis, M. Liakata, T. Damoulas, and A. I. Cristea. Can We Assess Mental Health through Social Media and Smart Devices? Addressing Bias in Methodology and Evaluation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 407–423. Springer, 2018.
- [242] A. Tsakalidis, M. Liakata, T. Damoulas, B. Jellinek, W. Guo, and A. Cristea. Combining Heterogeneous User Generated Data to Sense Well-Being. In *COLING*, pages 3007–3018, 2016.
- [243] A. Tsakalidis, S. Papadopoulos, A. I. Cristea, and Y. Kompatsiaris. Predicting Elections for Multiple Countries using Twitter and Polls. *IEEE Intelligent Systems*, 30(2):10–17, 2015.
- [244] A. Tsakalidis, S. Papadopoulos, and I. Kompatsiaris. An Ensemble Model for Cross-Domain Polarity Classification on Twitter. In *International Conference on Web Information Systems Engineering*, pages 168–177. Springer, 2014.
- [245] A. Tsakalidis, S. Papadopoulos, R. Voskaki, K. Ioannidou, C. Boididou, A. I. Cristea, M. Liakata, and Y. Kompatsiaris. Building and Evaluating Resources for Sentiment Analysis in the Greek Language. *Language Resources and Evaluation*, 52(4):1021–1044, 2018.
- [246] G. Tsebelis. Lessons from the Greek Crisis. *Journal of European Public Policy*, 23(1):25–41, 2016.
- [247] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welp. Predicting elections with Twitter: What 140 characters reveal about political sentiment. *ICWSM*, 10(1):178–185, 2010.

- [248] J. Turian, L. Ratinov, and Y. Bengio. Word Representations: A Simple and General Method for Semi-Supervised Learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics, 2010.
- [249] K. Tymoshenko, D. Bonadiman, and A. Moschitti. Convolutional Neural Networks vs. Convolution Kernels: Feature Engineering for Answer Sentence Reranking. In *NAACL-HLT*, pages 1268–1278, 2016.
- [250] C. Vania, M. Ibrahim, and M. Adriani. Sentiment Lexicon Generation for an Under-Resourced Language. *International Journal of Computational Linguistics and Applications*, 5(1):59, 2014.
- [251] S. Volkova, G. Coppersmith, and B. Van Durme. Inferring User Political Preferences from Streaming Communications. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 186–196, 2014.
- [252] D. T. Wagner, A. Rice, and A. R. Beresford. Device analyzer: Understanding smartphone usage. In *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*, pages 195–208. Springer, 2013.
- [253] B. Wang, M. Liakata, A. Tsakalidis, S. G. Kolaitis, S. Papadopoulos, L. Apostolidis, A. Zubiaga, R. Procter, and Y. Kompatsiaris. TOTEMSS: Topic-based, Temporal Sentiment Summarisation for Twitter. *Proceedings of the IJCNLP 2017, System Demonstrations*, pages 21–24, 2017.
- [254] B. Wang, M. Liakata, A. Zubiaga, and R. Procter. TDParse: Multi-target-specific sentiment recognition on twitter. In *EACL*, volume 1, pages 483–493, 2017.
- [255] R. Wang, M. S. Aung, S. Abdullah, R. Brian, A. T. Campbell, T. Choudhury, M. Hauser, J. Kane, M. Merrill, E. A. Scherer, et al. Crosscheck: toward passive sensing and detection of mental health changes in people with schizophrenia. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 886–897. ACM, 2016.
- [256] R. Wang, F. Chen, Z. Chen, T. Li, G. Harari, S. Tignor, X. Zhou, D. Ben-Zeev, and A. T. Campbell. Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 3–14. ACM, 2014.

- [257] D. Watson, L. A. Clark, and A. Tellegen. Development and validation of brief measures of positive and negative affect: the panas scales. *Journal of personality and social psychology*, 54(6):1063, 1988.
- [258] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics, 2005.
- [259] X. Zhu, S. Kiritchenko, and S. M. Mohammad. NRC-Canada-2014: Recent Improvements in the Sentiment Analysis of Tweets. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 443–447. Citeseer, 2014.
- [260] B. Zou, V. Lampos, and I. Cox. Multi-Task Learning Improves Disease Models from Web Search. In *Proceedings of the 2018 World Wide Web Conference*, pages 87–96. International World Wide Web Conferences Steering Committee, 2018.
- [261] B. Zou, V. Lampos, R. Gorton, and I. J. Cox. On Infectious Intestinal Disease Surveillance Using Social Media Content. In *Proceedings of the 6th International Conference on Digital Health Conference*, pages 157–161. ACM, 2016.
- [262] A. Zubiaga, A. Voss, R. Procter, M. Liakata, B. Wang, and A. Tsakalidis. Towards Real-time, Country-level Location Classification of Worldwide Tweets. *IEEE Transactions on Knowledge & Data Engineering*, (1):1–1, 2017.
- [263] A. Zubiaga, B. Wang, M. Liakata, and R. Procter. Stance Classification of Social Media Users in Independence Movements. *arXiv preprint arXiv:1702.08388*, 2017.

Appendices

APPENDIX A

Guidelines for the Manually Annotated Lexicon

A.1 Aim

Our aim is to create a manually annotated sentiment and emotion lexicon for the Greek language. By “sentiment and emotion lexicon” we mean the creation of a dictionary that will map specific words to some sentiment and emotional dimensions (e.g., subjectivity, positive/negative orientation, etc.). The lexicon will be primarily used to detect the sentiment and emotion expressed in social media posts (e.g., tweets).

A.2 Mining Triantafyllidis Lexicon

We have created a list of words that are likely to be sentiment- or emotion-loaded. To aggregate these words, we used the “advanced search” of the online version of Triantafyllidis Lexicon¹, searching for words that can be used in an ironic, derogatory, abusive, mocking or vulgar tone. We accompanied the retrieved words with others whose definitions in Triantafyllidis lexicon included one of the following words: συναίσθημα, αισθάνομαι, αίσθηση, αίσθημα, συναίσθηση, αισθάνεται, νιώθω. Overall, we aggregated 2,324 words. Our goal is now to annotate them with respect to the sentiment or emotion they might reveal when they are present within a sentence.

A.3 Annotation Guidelines: Subjectivity and Sentiment

Our annotations are based on the work by [258], who generated a lexicon consisting of words in the English language, annotated with respect to two dimensions:

- *Subjectivity Level* (strong, weak, none): Mapping of a word with respect to its subjectivity (strong or weak). The distinction between strong and weak subjectivity is based on the usage frequency: “Words that are subjective in most contexts were marked strongly subjective, and those that may only have certain subjective usages were marked weakly subjective” [258]. We have also allowed for annotating a word as non-subjective (“none”).

¹http://www.greek-language.gr/greekLang/modern_greek/tools/lexica/triantafyllides/index.html

- *Polarity* (positive, negative, both): Mapping each of the previously labelled “subjective” (either strong or weak) words to the sentiment they normally express, when present in a sentence (positive or negative). Since there might be ambiguous cases of words that are equally likely to be used in positive and negative contexts, we are also allowed to label a word as “both” (positive and negative). Note that in [258] only 0.3% of the words belonged in this category. Words that have not been labelled as “subjective” (i.e., those labelled as “none” with respect to their subjectivity label) should not be annotated with respect to their polarity or emotion (see next section).

A.4 Annotation Guidelines: Emotions

Besides the “traditional” subjectivity and polarity detection tasks, we are interested in annotating further emotional dimensions of the words. For this reason, we have added six further emotional dimensions, based on Ekman’s basic emotions (anger, disgust, fear, happiness, sadness and surprise) [68]. Our goal is to annotate every word with respect to each of these emotions. In particular, our goal is to annotate a specific word with respect to each emotion emotion, depending on the level (1 to 5) to which its presence in a sentence could be indicative of the user (i.e., author of the message) expressing this emotion. For example, if the word “wow” is present in a sentence, then it is highly likely that this sentence is expressing the *surprise* of the user who wrote it; thus, the word “wow” would be annotated as highly indicative of the “surprise” emotion and we would annotate the corresponding dimension as 5.

Finally, we have added two extra columns – one for annotating every word with respect to its part-of-speech and another for adding any comments on our annotation or proposing similar words to the one we have annotated.

A.5 Lexicon Use

Our initial goal is to use the annotated lexicon for analysing user-generated content from social media. In particular, we would like to test if such a lexicon can be employed to detect subjectivity in tweets written in the Greek language and if it can be used as a “psychometric” tool for monitoring real-world events. Finally, we aim at releasing our lexicon so that it can be used by future work in the domain of sentiment analysis in the Greek language, and be potentially enriched with further annotations by members of other research organisations and institutions.

APPENDIX B

List of Keywords

The list of keywords that were used in Chapter 5 to aggregate politically-related tweets using Twitter Streaming API was the following (per country):

Germany: @CDU, @CSU, CDU, CSU, Christlich Demokratische Union Deutschlands, Christlich Demokratische Union, Christlich Soziale Union in Bayern, Christlich Soziale Union, @spdde, Sozialdemokratische Partei Deutschlands, Sozialdemokratische Partei, SPD, Sozialdemokratische Partei, @dieLinke, Die Linke, Linke, @Die.Gruenen, Die Grünen, Die Grunen, Grüne, Grune, Bündnis 90, Bündnis 90, @fdp, Freie Demokratische Partei, FDP, @AfD_Bund, Alternative für DE, Alternative für DE, AfD, Alternative für Deutschland, Alternative für Deutschland

The Netherlands: Partij voor de Vrijheid, PVV, @VVD, Volkspartij voor Vrijheid en Democratie, VVD, @D66, Democraten 66, D66, @cdavandaag, Christen Democratisch Appél, Christen Democratisch Appel, CDA, @PvdA, Partij van de Arbeid, PvdA, @SPnl, Socialistische Partij, SP, @christenunie, ChristenUnie, CU, @SGPnieuws, Staatkundig Gereformeerde Partij, SGP, @groenlinks, GroenLinks, @50pluspartij, 50PLUS, 50+, @PartijvdDieren, Partij voor de Dieren, PvdD

Greece: Νέα Δημοκρατία, Νεα Δημοκρατια, ΝΔ, Ν.Δ., ΣΥΡΙΖΑ, Σύριζα, ΣΥ.ΡΙΖ.Α., Συνασπισμός Ριζοσπαστικής Αριστεράς, Συνασπισμος Ριζοσπαστικής Αριστερας, Ελιά, Ελια, Ποταμι, Ποτάμι, Χρυσή Αυγή, Χρυση Αυγη, Χ.Α., ΧΑ, ΚΚΕ, Κ.Κ.Ε., Κομμουνιστικό Κόμμα Ελλάδας, Κομμουνιστικο Κομμα Ελλαδας, Ανεξάρτητοι Έλληνες, Ανεξαρτητοι Ελληνες, Αν.Ελ., Ανέλ, ANEL, Δημάρ, ΔΗΜΑΡ, ΔΗΜ.ΑΡ., Δημοκρατική Αριστερά, Δημοκρατικη Αριστερα, @neademokratia, @syriza-gr, @DParataxi, @ToPotami, @xryshaygh, @anexartittoi, @dimokratiki